DOCUMENT RESUME

ED 239 584 IR 010 946

AUTHOR Adams, David R., Ed.; Athey, Thomas H., Ed.
TITLE DPMA Model Curriculum for Undergraduate Computer Information Systems Education.
INSTITUTION Data Processing Management Association Education Foundation, Park Ridge, IL.
PUB DATE 82
NOTE 72p.; Prepared by the Education Foundation Committee on Curriculum Development.
PUB TYPE Guides - Classroom Use - Guides (For Teachers) (052) -- Reports - Descriptive (141)

EDRS PRICE MF01/PC03 Plus Postage.
DESCRIPTORS *Business Education; Computer Programs; *Computers; *Computer Science Education; Course Descriptions; *Curriculum Development; Databases; Guidelines; Higher Education; *Information Systems; Models; *Programing; Undergraduate Study

ABSTRACT

Designed primarily for 4-year undergraduate programs for business applications programmer/analysts offered through schools of business or through applied computer science programs that require a concentration of business courses, these guidelines are based on national and regional conferences, questionnaire surveys, and consultation with computer experts from education and industry. A general discussion covers the model curriculum development process, the role of computer information systems (CIS), background for CIS careers, and the technical environment of CIS. A summary of the CIS curriculum includes an overview, a list of objectives, and a brief description of the core courses, curriculum structure, elective courses and business support courses. Detailed specifications are given for each of the 15 curriculum courses: (1) Introduction to Computer Based Systems; (2-3) Applications Program Development (I) and (II); (4) Systems Analysis Methods; (5) Structured Systems Analysis and Design; (6) Database Program Development; (7) Advanced Database Concepts; (8) Applied Software Development Project; (9) Software and Hardware Concepts; (10) Office Automation; (11) Decision Support Systems; (12) Distributed Data Processing; (13) EDP Audit and Controls; (14) Information Systems Planning; and (15) Information Resource Planning. Included are course descriptions, major student outcomes expected, course content, teaching approaches, and recommended resource materials and references. (LMM)

****
* Reproductions supplied by EDRS are the best that can be made *
* from the original document. *
****

Data Processing Management Association Education Foundation
505 Busse Highway, Park Ridge, IL 60068

U.S. DEPARTMENT OF EDUCATION
NATIONAL INSTITUTE OF EDUCATION
EDUCATIONAL RESOURCES INFORMATION CENTER (ERIC)

This document has been reproduced as received from the person or organization originating it.

Minor changes have been made to improve reproduction quality.

• Points of view or opinions stated in this document do not necessarily represent official NIE position or policy.

ED239584

DPMA Model Curriculum for Undergraduate Computer Information Systems Education

"PERMISSION TO REPRODUCE THIS MATERIAL HAS BEEN GRANTED BY

_____

_____

TO THE EDUCATIONAL RESOURCES INFORMATION CENTER (ERIC)."

IR010974

Prepared by the DPMA Education Foundation Committee on Curriculum Development

DPMA MODEL CURRICULUM FOR UNDERGRADUATE COMPUTER INFORMATION SYSTEMS EDUCATION

David R. Adams
and
Thomas H. Athey
Editors

Prepared by the DPMA Education Foundation Committee on Curriculum Development

Education Foundation
Data Processing Management Association
505 Busse Highway
Park Ridge, Illinois 60068

Copyright © 1981. Data Processing Management Association Education Foundation, 505 Busse Highway, Park Ridge, IL 60068 USA. ALL RIGHTS RESERVED. No part of this book may be reproduced by any means, electronic, mechanical, photocopying, recording or otherwise, nor transmitted, nor translated into a machine language for storage and/or retrieval, without the written permission of the DPMA Education Foundation, except for brief excerpts to be used in published reviews of the book.

Second printing, 1982.

IR010946

CONTENTS

	Page
Preface	1
The Curriculum Development Process	4
The Role of Computer Information Systems	6
Background of CIS Careers	8
The Technical Environment of CIS	9
The CIS Curriculum	10
Curriculum Objectives and Overview	10
CIS Core Courses	11
CIS Elective Courses	12
Business Support Courses	13
CIS Curriculum Structure	15
CIS Course Specifications	17
CIS-1 Introduction to Computer Based Systems	18
CIS-2 Application Program Development I	22
CIS-3 Application Program Development II	26
CIS-4 Systems Analysis Methods	30
CIS-5 Structured Systems Analysis and Design	33
CIS-6 Database Program Development	36
CIS-7 Applied Software Development Project	40
CIS-8 Software and Hardware Concepts	42
CIS-9 Office Automation	45
CIS-10 Decision Support Systems	48
CIS-11 Advanced Database Concepts	51
CIS-12 Distributed Data Processing	54
CIS-13 EDP Audit and Controls	56
CIS-14 Information Systems Planning	60
CIS-15 Information Resource Management	64

DPMA MODEL CURRICULUM FOR UNDERGRADUATE COMPUTER INFORMATION SYSTEMS EDUCATION

Preface

This report contains curriculum guidelines for undergraduate programs in Computer Information Systems (CIS). It is the culmination of over two years of effort in studying the academic preparation required by persons who will enter the field of information systems as business application programmer/analysts and who will grow professionally within normal career paths open to business computer specialists. Undergraduate programs with the goal of preparing application programmer/analysts for commercial information system environments are defined in this report as Computer Information Systems programs.

These curriculum guidelines are designed primarily for four-year undergraduate programs offered either through schools of business or through applied computer science programs which require a concentration of business courses in support of the technically oriented computer courses. The recommendations also have implications for two-year community college curriculums in data processing where transfer credit towards CIS programs is offered. These guidelines recognize the need for articulation between two and four-year programs, as well as the value of having terminal degree courses of study at both levels.

The model curriculum presented in this report is the result of the combined efforts of computer educators, professional computer specialists, and data processing and information systems managers to identify the need for commercial programmer/analysts, to define the education and training requirements for servicing those needs, and to specify those requirements as a formal set of undergraduate coursework. The curriculum evolved through two national conferences, six regional conference/workshops, two national questionnaire surveys, and numerous consultations with computer, business, and education experts. The curriculum development process was nationwide in scope with extensive academic/industry involvement and with joint support for its recommendations.

It should be emphasized that the model curriculum is a "living" document, the product of a development process designed to incorporate new requirements and changes that will be inevitable in a dynamic field such as computer education. Thus, this document is not intended as the final work on CIS education, nor are its recommendations transitory. The guidelines provide the stability of foundational knowledge in Computer Information Systems, yet the mechanisms are in place to ensure thoughtful evolution of the curriculum as changes become necessary.

This report was prepared by the Data Processing Management Association (DPMA) Education Foundation Committee on Curriculum Development. Members of this committee are:

Thomas H. Athey (Chairman), California State Polytechnic University, Pomona
David R. Adams, Northern Kentucky University
Charles F. Durang, Winthrop Publishers, Inc.
Donald E. Price, Sierra Community College
James Stallard, General Dynamics Corporation
Gerald Wagner, California State Polytechnic University, Pomona

Course specifications appearing in the report were prepared by project teams under the direction of the following persons:

CIS-1 INTRODUCTION TO COMPUTER-BASED SYSTEMS
Robert Behling, Boise State University

CIS-2 APPLICATIONS PROGRAM DEVELOPMENT I
CIS-3 APPLICATIONS PROGRAM DEVELOPMENT II
Don Medley, Moorpark Community College

CIS-4 SYSTEMS ANALYSIS METHODS
CIS-5 STRUCTURED SYSTEMS ANALYSIS AND DESIGN
Michael Powers, Illinois State University

CIS-6 DATABASE PROGRAM DEVELOPMENT
CIS-11 ADVANCED DATABASE CONCEPTS
Chris Pidgeon, California State Polytechnic University, Pomona

CIS-7 APPLIED SOFTWARE DEVELOPMENT PROJECT
Claude Simpson, Jr., Northeast Louisiana University

CIS-8 SOFTWARE AND HARDWARE CONCEPTS
Robert Zant, North Texas State University

CIS-9 OFFICE AUTOMATION
Roy A. Boggs, University of North Carolina, Greensboro

CIS-10 DECISION SUPPORT SYSTEMS
John F. Schrage, Southern Illinois University, Edwardsville

CIS-12 DISTRIBUTED DATA PROCESSING
Herbert W. Bomzar, Central Michigan University

CIS-13 EDP AUDIT AND CONTROLS
Frederick Gallegos, U.S. General Accounting Office

CIS-14 INFORMATION SYSTEMS PLANNING
Vince E. Heiker, Boise Cascade

CIS-15 INFORMATION RESOURCE MANAGEMENT
Ivan Gavrilovic, IBM Systems Research Institute

Persons who made meaningful contributions to this project as members of course development teams were:

David G. Barber, University of Arkansas
Martin L. Bariff, Case Western Reserve University
Barbara Beccue, Bradley University
Charles P. Bilbrey, James Madison University
N. D. Brammer, Colorado State University
Ralph F. Brueggemann, Cincom Systems, Inc.
Marvin C. Chaiken, Union Mutual Insurance Co.
Frank Clamon, Walters State Community College
W. K. Daugherty, New Mexico State University
Richard L. Diamond, Verbatim Corporation
Felix Dreher, Kansas State University of Pittsburg
Carl Everett, Xavier University
Melvin Franz, Central Missouri State University
George Goldstein, Miami-Dade Community College
Doris D. Gottschalk, California State University, Hayward
Robert T. Grauer, University of Miami
Gary Green, Arizona State University
Stephen Harvey, Missouri Department of Revenue
Herman P. Hoplin, Syracuse University
Sid Huff, The University of Western Ontario
Eleanor Jordan, University of Texas
David M. Kroenke, Consultant
Richard Kuhr, Bendix Corporation
George Lanphere, Associates Computer Services
F. Hollis Latimer, Tarrant County Jr. College
Alan Lefko, Eastman Kodak Company
Duane F. Lingenfelter, FMC Corporation
Alden C. Lorents, Northern Arizona University
Thomas Luce, Odessa College
Roy Martin, West Texas State University
Emerson C. Maxson, Boise State University
Joseph McPadden, Computer Sciences Corporation
D. M. Miller, University of Winnipeg
P. K. Misra, Ralston Purina Company
Don Mitchell, Arkansas Technical University
B. E. Moody, Quinnipiac College
Judson Ostle, Hewlett-Packard
V. Arthur Owles, Illinois State University
Robert A. Palgon, Western Electronic
Stan Piercefield, American Fletcher National Bank
David E. Pope, Shell Oil Company
Herbert Rebhun, University of Houston
Jerome Rolefson, University of Wisconsin, Milwaukee
Fred Rye, University of Texas

Davis Sellers, Oklahoma State Tech
Blair Stephensen, Pegasus Systems Assoc.
Nancy Stern, Hofstra University
Robert Stumpf, California State Polytechnic Univ., Pomona
Ralph A. Szweda, Monroe Community College
Lavette Teague, California State Polytechnic Univ., Pomona
William Tryon, Monsanto Company
Shelby R. Tunmine, Texas Farm Bureau Insurance
Barbara Venz, William Woods College
Roy F. Waller, Western Electric Company
Ira Weiss, University of Houston
David C. Whitney, San Francisco State University
G. W. K. Willis, University of Texas, San Antonio
Bruce Zak, LWFW, Inc.

Other readers and evaluators of the curriculum recommendations were:

Robert Bigalow, Computer Law Service
Tom Cashman, Long Beach City College
Phillip Gensler, West Texas State University
Steve Mitchell, INTERFACE, Computer Education Quarterly
Gary Shelly, Anaheim Publishing Company

In addition, there were over 400 other persons who answered questionnaires, contributed in workshop sessions, or otherwise participated in this project.

Regional conference organizers were Tom Murphy and John Lehane (San Francisco), Terrence J. Boyer and Bob Benson (St. Louis), Herb McClendon and Michael Troxell (Dallas), Steve Vajda and Charles Stratton (Miami), Ralph Jones, Peter Hanna, and Marv Wofsey (Washington D. C.), and Al Orsava and Florence Murray (Toronto).

The Curriculum Development Process

The efforts which have culminated in this report originated with the Information Systems Department of California State Polytechnic University, Pomona. In February, 1979, the Department hosted the first national conference/workshop on Computer Information Systems education, bringing together representatives of industry and education to discuss and to make recommendations for improving computer education at the undergraduate level. This conference established (1) that there are significant differences among educational programs in information systems, computer science, and computer engineering, (2) that programs in computer information systems have the distinct primary goal of preparing entry-level applications programmer/analysts for commercial environments, and (3) that a nationally recognized model curriculum offering guidance in the development of educational programs in computer information systems be established. The conference concluded with the formation of a task group to develop specific plans for establishing this model curriculum.

The success of this first conference and the potentially beneficial impact of a model curriculum on information systems education attracted the attention of the Data Processing Management Association (DPMA). At a task group meeting in June, 1979, DPMA expressed its interest in co-hosting with Cal Poly University a second national conference to continue development of the curriculum. It had become apparent by that time that there was widespread interest in the issues addressed at the first conference and a groundswell of enthusiasm among data processing educators and professionals to see continued progress made in clarifying the role of computer information systems education.

The second national conference/workshop was held in January, 1980. It was attended by computer educators and industry representatives from across the nation. Discussion centered on a preliminary version of the model curriculum developed as an outgrowth of the first conference. Through workshop sessions this initial draft was critiqued and refined and a revised version developed. This revised curriculum identified a set of core courses that was agreed should form the basis for all undergraduate programs in Computer Information Systems. Future efforts were to be geared to defining these courses in greater detail and in gaining consensus on the recommendations among as many educators and computer professionals as desired to participate. As a result of this very successful second conference, a commitment was made by the DPMA Education Foundation to sponsor the curriculum development project and to support its recommendations as viable standards for Computer Information Systems education.

Following this conference, a DPMA Committee on Curriculum Development was formed to oversee the project, and a target date of June, 1981, was tenta- tively set for its completion. The Committee adopted as its overall objec- tive "to develop a nationally recognized and accepted model curriculum for undergraduate Computer Information Systems education."

The preliminary version of the curriculum and a questionnaire soliciting comments about it were sent to educators and industry representatives across the country. During the summer of 1980, the Curriculum Committee reviewed the survey results and revised the curriculum in preparation for presenting the draft at a series of one-day conference/workshops to be held during the following year at various college and industry sites around the country.

Five such regional conferences were held between October, 1980, and April, 1981, in St. Louis, Dallas, Miami, Washington, D.C., and Toronto, Canada. These one-day workshops allowed persons who were unable to attend the national conferences to take part in the curriculum development process. Using the curriculum core developed at the national conferences and refined through the questionnaire, workshop participants studied and critiqued the ongoing revisions of these core courses and the set of recommended elective courses that were evolving. In the meantime, a second questionnaire had been distributed to a growing list of interested participants, and their reactions and suggestions were embodied in the various iterations of the curriculum.

During January, 1980, 12 project leaders were appointed to oversee development of the descriptions and specifications for all the courses in the curriculum. Each course development team was comprised of from five to nine persons, chosen from among educators, authors, and data processing professionals and managers. These groups were charged with the responsibility for detailed design of the courses--preparing outlines of the subject matter, suggesting teaching/learning methodologies, and providing suggested reference materials for use in the courses. The specifications prepared by these groups became the primary points of discussion at the regional conferences and were amended accordingly. Members of the Curriculum Committee served to make sure that the various revisions were philosophically and substantively consistent with the original intent of the curriculum development project.

In May, 1981, a draft of the curriculum report was presented at the National Computer Conference in Chicago; by the middle of June, the curriculum was in its final stages. Members of the Curriculum Committee prepared the final version, integrating the course specifications from the course development teams, the comments received from academic and industry reviewers, and the recommendations coming from the conferences and questionnaire surveys. This report is the product of that extensive and intensive curriculum development process. The model curriculum that surfaced from that process represents the efforts of many people. It portrays the fundamental knowledge, abilities, and attitudes required for education in Computer Information Systems, tempered by the employment realities demanded in the information processing profession.

The Role of Computer Information Systems

Computer Information Systems is an emerging academic discipline with goals, subject matter, and problem-solving processes sufficiently different from other computer-related disciplines to warrant special attention. To understand those differences it is convenient to conceptually divide the computer disciplines into two broad categories: those that educate developers of basic computer technology and those that educate users of that technology. In the former category are the traditional disciplines of computer engineering and computer science; in the latter category is the growing area of computer information systems.

Computer engineering deals primarily with the development of computer hardware technology. Computer engineers are educated in the design and fabrication of hardware components of computer systems and in the development of circuit logic to carry out the basic operations of those components. Typically, they are schooled in departments of computer engineering, and seek employment with computer and peripheral device manufacturers. Their education includes coursework in electrical engineering, mechanical engineering, mathematics, and computer science.

Computer science, on the other hand, is concerned mainly with the development of computer system software technology. Computer scientists are educated in the design and implementation of system software--operating systems, language translators, data management software, and other programming, processing, and operating aids that facilitate use of the computer hardware. Their education has a strong theoretical, scientific, and mathematical emphasis, and graduates of computer science programs commonly seek employment with computer manufacturers or software houses that specialize in system software.

Thus, the graduates of computer engineering and computer science programs become developers of basic computer hardware and system software technology. They provide a useful mechanism--the computer--which can be applied to the solutions of various problems in many different fields. Scientific and business applications represent two of the major areas to which computers are applied.

Scientific, engineering, and/or mathematical applications of computers demand some of the same abilities as are emphasized in computer engineering and computer science programs. Historically, the ranks of scientific programmers and systems developers have often been populated by graduates of these types of programs.

Within the business applications area there have been three distinct, yet often overlapping, educational thrusts: business data processing, management information systems (MIS), and computer information systems. Business data processing programs are offered at the community college level and have the purpose of training applications programmers for commercial environments. MIS programs are offered at the baccalaureate level and prepare persons for systems analysis and information systems management careers. CIS programs offer four-year technical/business education to prepare business applications programmer/analysts.

Although community college programs vary widely in the number and types of courses offered, many have the objectives of preparing business programmers. The primary emphasis in these business data processing curriculums is in programming methodology and programming languages. Usually, secondary emphasis is given to such areas as systems analysis and design, computer operations, systems documentation, and business theory. Graduates of these types of programs often seek employment as commercial programmers.

Management information systems programs are targeted for careers in information analysis, systems design, and information systems management. There is strong emphasis on business and organizational studies as well as information systems technology. Coursework typically includes topics such as organization theory, decision theory, quantitative business analysis, information analysis, systems design, data administration, and information systems policy at the corporate level. Secondary emphasis is given to technical computer studies. MIS curriculums are designed to be offered through schools of business as adjuncts to the core business coursework recommended by the American Assembly of Collegiate Schools of Business (AACSB), the accrediting agency for business programs.

Computer Information Systems programs are located in schools of business or in applied computer science departments and are primarily concerned with the application of the system development life cycle to business-oriented, computer-based systems. As such, their subject matter includes the study of systems analysis, systems design, and computer programming, along with other technical and business areas pertinent to the development, implementation, and maintenance of information systems in a variety of organizational settings. Unlike MIS programs with their "business plus information systems" emphasis, CIS programs are more technically oriented, emphasizing programming and systems development skills along with information systems and business understanding.

The goal of the CIS course of study is to prepare the business applications programmer/analyst (or, in some organizations called analyst/programmer) who is well-schooled in the technical aspects of computer systems, is knowledgeable of the applications area to which they are applied, and has sufficient background to be able to grow professionally in a rapidly changing field. CIS programs are career oriented, with the expectation that these programmer/analysts will have a sufficiently rigorous background that they can, with additional experience, advance along several career paths leading to positions in systems analysis, systems design, programming and systems project leadership, and systems management.

Background for CIS Careers

The primary thrust of educational programs in Computer Information Systems is in enabling graduates to function as applications programmer/analysts in commercial environments. These graduates need competencies along all dimensions of systems development, requiring a greater range of skills and understandings than do persons who will function exclusively as programmers or analysts. Competency requirements for programmer/analysts include:

1. being acquainted with the full systems development life cycle;

2. being able to function effectively in the planning, systems analysis and design, and program design, development, testing, and maintenance stages of the life cycle;

3. being able to program in depth in at least one higher-level programming language;

4. being able to work with users of information systems in developing computer solutions to business problems;

5. being able to communicate ideas and project results orally and in written form;

6. being able to function individually or as a member of a project team;

7. being acquainted with the technical and human aspects of change;

8. being acquainted with the basic principles and practices of managing people and tasks through project management;

9. being acquainted with business functions and operations and understanding the role of computer information systems in the business enterprise; and

10. having the education and background necessary to continue with lifelong learning and personal development in addition to becoming technically competent.

Such needed competencies dictate that the programmer/analyst receive education and training in at least three different areas--(1) in systems development methodologies, which provide the fundamental problem-solving approaches used in the profession; (2) in technical computer skills, which provide the tools for implementing those problem solutions; and (3) in business theory, which provides an understanding of the context within which the systems are implemented.

The Technical Environment of CIS

Programmer/analysts who enter industry in the 1980's will encounter a dramatically changing technical environment which will demand changes in the way systems development has been carried out in the past. In particular, several trends appear evident and will impact the profession.

First, the field of computer information systems is moving rapidly towards transaction-oriented, terminal-based, database systems. Data communications and distributed data processing are becoming integral parts of many information systems as data entry and data processing are moved out of the computer room and placed in the hands of non-technical users. Systems to implement these types of applications are more complex and require increased consideration for human/software/hardware interfaces than did their batch processing counterparts. Mini/micro computers are becoming equally important as mainframe computers in supporting these systems.

Second, with increasingly complex software, the requirement for more sophisticated development tools becomes paramount. Ad hoc programming methods are giving way to integrated methodologies. Structured systems development strategies and techniques are becoming primary tools in the programmer/analyst's repertoire and greater emphasis is being given to the "front end" of the system development process-to the problem analysis, of system design, and program design phases. Automated development tools and other productivity aids are being used with greater frequency to increase programmer/analyst productivity and the effectiveness of the systems they develop and maintain.

Third, there is a trend away from the use of assembly-level languages towards higher-level procedural and non-procedural languages. COBOL will likely remain as the dominant business application language, especially on main frame computers, yet improved methods of programming will increase the effectiveness of the COBOL programmer. Other mini- and microcomputer languages will continue in development and "user friendly" inquiry, report writer, and decision support languages will become increasingly popular ways for non-technical persons to use the computer. The programmer/analyst will be required to develop new software using these higher-level languages and improved methodologies and will have to know how to integrate this original software with packaged applications. The ability to consult in "make or buy" decisions will become an important skill for the programmer/analyst.

Fourth, the kinds of business problems to which computers will be applied will expand. Not only will replacement systems to upgrade existing batch and on-line systems in the data processing area be required, but also new systems that aid in the decision making process and that automate the office function will be developed. The computer will become a management tool as well as an operational tool and will be used in the planning, control, and decision making activities of the organization. The implications for the programmer/analyst are for greater understanding of the business processes that are computerized and for more sophisticated technical knowhow and tools for dealing with the complexities of these systems.

The CIS Curriculum

Curriculum Objectives and Overview

The primary objective of the DPMA Model Curriculum for Computer Information Systems is:

"to provide graduates with the knowledge, abilities, and attitudes to function effectively as applications programmer/analysts, and with the educational background and desire for lifelong professional development."

Specific curriculum objectives that contribute to this overall objective are:

1. to provide understanding of the goals, functions, and operations of business organizations;

2. to provide understanding of the information needs and the role of information systems in these organizations;

3. to provide the analytical and technical skills for identifying, studying, and solving information problems within organizations;

4. to provide communications and human relations skills for effective interaction with organization members, especially with the users and developers of information systems;

5. to provide knowledge and ability for effective management of information systems projects;

6. to instill a professional attitude and seriousness of purpose about Computer Information Systems as a career field; and

7. to provide the background for further study of and professional advancement in the field of Computer Information Systems.

The model curriculum presented here operationalizes these objectives through an integrated set of courses corresponding generally to the major activities of the systems development life cycle, along with other courses dealing with the hardware, software, applications, organizational, managerial, and business aspects of systems development. The curriculum provides the technical/business competencies for entry-level success, for adaptation to changing requirements in the technical and business environments, and for advancement in the information processing profession.

The CIS model curriculum is comprised of:

1. seven required CIS core courses,

2. three other CIS courses to be chosen from a set of eight recommended elective courses, plus

3. a minimum set of business support courses.

This course of study represents the minimum technical and business requirements for offering Computer Information Systems programs. All CIS courses are designed to be taught as 3-unit, semester-long courses for a total of 30 semester hours.

Within a baccalaureate degree program there should be offered a set of general education courses--in the arts, sciences, and humanities--to broaden intellectual awareness and to serve in the development of cultural literacy and analytical and evaluative methods.

CIS Core Courses

The curriculum core is comprised of the following four lower-division (freshman/sophmore level) courses and three upper-division (junior/senior level) courses.

CIS-1 INTRODUCTION TO COMPUTER-BASED SYSTEMS.
An introduction to computers and data processing taught as a general education course for all students (lower division).

CIS-2 APPLICATIONS PROGRAM DEVELOPMENT I.
A beginning computer problem solving and programming course using COBOL as the vehicle language (lower division).

CIS-3 APPLICATIONS PROGRAM DEVELOPMENT II.
An advanced computer problem solving and programming course using COBOL (lower division).

CIS-4 SYSTEMS ANALYSIS METHODS.
An overview of the systems development life cycle with emphasis on techniques and tools of system documentation and logical system specifications (lower division).

CIS-5 STRUCTURED SYSTEMS ANALYSIS AND DESIGN.
Advanced coverage of the strategies and techniques of structured systems development (upper division).

CIS-6 DATABASE PROGRAM DEVELOPMENT.
A course emphasizing software design and programming in a database environment (upper division).

CIS-7 APPLIED SOFTWARE DEVELOPMENT PROJECT.
A capstone systems course integrating the knowledge and abilities gained through the other computer-related courses in the curriculum within a comprehensive system development project (upper division).

The lower-division courses may be taught at the community college level with credit directly transferrable into a four-year CIS program. To facilitate transfer of knowledge between these courses and upper-division courses, community college programs should adhere to the course specifications presented in this report.

CIS Elective Courses

In addition to the seven core courses, three upper-division elective courses are required. These electives should be chosen from the following set of courses:

CIS-8 SOFTWARE AND HARDWARE CONCEPTS.
A survey of technical topics related to computer systems with emphasis on the relationships between hardware architecture, systems software, and applications software.

CIS-9 OFFICE AUTOMATION.
An examination of the office as a center of business activity, operational logistics, and decision support, and the impact of automation on the office environment.

CIS-10 DECISION SUPPORT SYSTEMS.
An analysis of the highest level of information support systems which aid the manager in the decision-making process.

CIS-11 ADVANCED DATABASE CONCEPTS.
An in-depth investigation of data modeling, system development, and data administration in a database environment.

CIS-12 DISTRIBUTED DATA PROCESSING.
An examination of the features and impact of distributed systems in the business enterprise.

CIS-13 EDP AUDIT AND CONTROLS.
An introduction to EDP auditing with emphasis on EDP controls, audit types, and audit techniques and their effects on system development.

CIS-14 INFORMATION SYSTEMS PLANNING.
An introduction to the financial, technical, and strategic information systems planning process.

CIS-15 INFORMATION RESOURCE MANAGEMENT.
A seminar in information systems management with emphasis on planning, organizing, and controlling user services and managing the systems development process.

It is recognized that curriculum flexibility is required in order for a school to adapt to local needs and circumstances. Therefore, the set of elective courses listed above are not meant to be exhaustive of all possible courses, but only identify those from which the three CIS elective courses should be chosen and which should be taught similarly and be directly transferrable for credit among different institutions. Additional elective courses that satisfy the mission of the school are encouraged with the recommendation that transfer credit for such courses be handled on a case-by-case basis.

Business Support Courses

An integral part of the Computer Information Systems curriculum is the set of recommended business support courses. These courses provide understanding of the context of the system development process--of the purposes and principles of operation of the functional areas of business. Such knowledge is vital in the effective analysis, design, and development of information systems to serve the operational, managerial, and decision-making needs of the organization.

The recommended business support courses are:

BUS-1 FINANCIAL ACCOUNTING PRINCIPLES
Coverage of the accounting cycle and the preparation of primary financial statements.

BUS-2 MANAGERIAL ACCOUNTING PRINCIPLES
Accounting concepts for internal use of management in planning and control of operations.

BUS-3 QUANTITATIVE METHODS
Application of quantitative approaches and techniques to solving decision problems.

BUS-4 PRINCIPLES OF MANAGEMENT
Basic concepts of systems, decision making, organizational evolution, and interpersonal considerations within organizations, and the planning, organizing, staffing, and control activities of managers.

BUS-5 PRINCIPLES OF MARKETING
Study of flow of goods from producer to consumer--the demand-stimulated and demand-fulfilling activities of business enterprises.

BUS-6 PRINCIPLES OF FINANCE
Study of the demand for funds and supply of funds by business, consumers, government, and foreign sectors; financial institutions, instruments, and policies.

BUS-7 ORGANIZATIONAL BEHAVIOR
Human aspects of organizations; individual and interpersonal behavior and group processes and their effects on the organization.

BUS-8 PRODUCTION AND OPERATIONS MANAGEMENT
Concepts underlying effective operating control of manufacturing and service businesses; approaches to production control, inventory policy, facilities planning, methods improvement, and technological assessment.

These are the minimum required courses to support the CIS curriculum. At schools that are accredited by AACSB, the common body of knowledge (cbk) in business satisfies these criteria. In non-accredited business schools and within schools where the CIS program is administered outside of business, these or similar sets of courses that implement a core of business knowledge should be required.

CIS Curriculum Structure

Figure 1 shows the relationships between the required, the recommended elective, and the business support courses. Connecting lines in the diagram indicate progression through the coursework and prerequisite courses. Note that courses CIS-1 through CIS-4 are lower-division courses and CIS-5 through CIS-15 are upper-division courses. The hierarchy implied in the diagram is roughly equivalent to the eight-semester sequence of undergraduate schooling.

Figure 1. Structure of DPMA Model Curriculum for Computer Information Systems

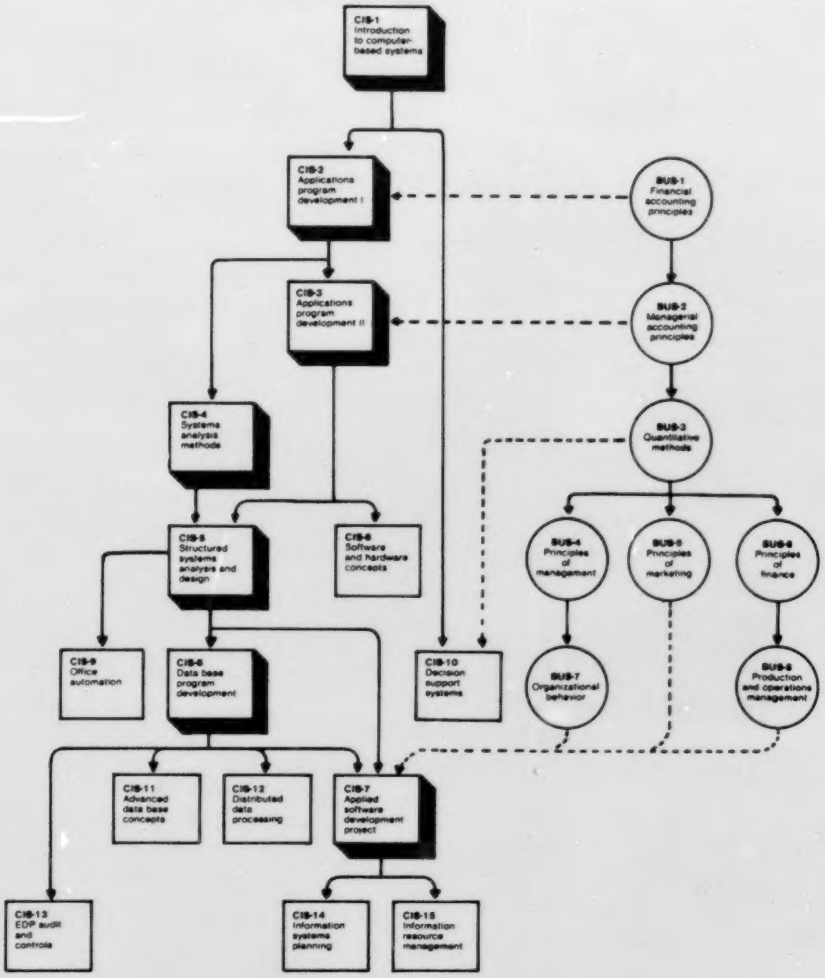
CIS-1 Introduction to computer-based systems
CIS-2 Applications program development I
CIS-3 Applications program development II
CIS-4 Systems analysis methods
CIS-5 Structured systems analysis and design
CIS-6 Software and hardware concepts
CIS-8 Data base program development
CIS-9 Office automation
CIS-10 Decision support systems
CIS-11 Advanced data base concepts
CIS-12 Distributed data processing
CIS-7 Applied software development project
CIS-13 EDP audit and controls
CIS-14 Information systems planning
CIS-15 Information resource management
BUS-1 Financial accounting principles
BUS-2 Managerial accounting principles
BUS-3 Quantitative methods
BUS-4 Principles of management
BUS-5 Principles of marketing
BUS-6 Principles of finance
BUS-7 Organizational behavior
BUS-8 Production and operations management

Table 1 shows the recommended courses distributed throughout the four-year program.

Table 1. Courses in the DPMA Computer Information Systems Model Curriculum

	CIS core courses	CIS elective courses*	Business support courses**	General Education courses
Freshmen/ Sophomore level	CIS-1. Introduction to computer-based systems CIS-2. Applications program development I CIS-3. Applications program development II CIS-4. Systems analysis methods		BUS-1. Financial accounting principles BUS-2. Managerial accounting principles	Arts Sciences Humanities
Junior/ Senior level	CIS-5. Structured systems analysis and design CIS-6. Data base program development CIS-7. Applied software development project	CIS-8. Software and hardware concepts CIS-9. Office automation CIS-10. Decision support systems CIS-11. Advanced data base concepts CIS-12. Distributed data processing CIS-13. EDP audit and control CIS-14. Information systems planning CIS-15. Information resource management	BUS-3. Quantitative methods BUS-4. Principles of management BUS-5. Principles of marketing BUS-6. Principles of finance BUS-7. Organizational behavior BUS-8. Production and operations management	

*A minimum of three elective courses chosen from this lisst.
**These eight or comparable courses meet minimum requirements.

CIS Course Specifications

The remainder of this report contains specifications for each of the 15 courses in the Computer Information Systems curriculum. Included in these specifications are course descriptions, major student outcomes expected, course contents, approaches to teaching the courses, and recommended resource materials and references.

The content of each course is categorized in major subject areas; however, there is no implication that the topics are to be covered in the sequence in which they are presented. The purpose of the outline is to suggest pertinent topics, skill levels, and proportions of time to be devoted to each. The time percentages refer to in-class presentations and discussions of the subject matter. There is the expectation that additional out-of-class work will be required. Recommended skill-level categories for the courses are:

SKILL LEVEL 1 - the student is being exposed to the subject matter for the first time.

SKILL LEVEL 2 - the student studies and/or applies the subject matter at a medium level of rigor and complexity.

SKILL LEVEL 3 - the student studies and/or applies the subject matter at a job-entry level of rigor and complexity.

CIS-1 INTRODUCTION TO COMPUTER-BASED SYSTEMS

COURSE DESCRIPTION

An overview of computer information systems. This survey course introduces computer hardware, software, procedures, systems, and human resources and explores their integration and application in business and in other segments of society. The fundamentals of computer problem solving and programming in a higher-level programming language are discussed and applied.

STUDENT OUTCOMES

To prepare business majors and others to be intelligent users of computers and to understand the basics of successful computer information systems: programs, procedures, data, people, computer hardware, computer programming in a higher-level language, and computer applications in various segments of society.

COURSE CONTENT

1. Introduction to Computer Information Systems (10%) — Skill Level 1

   Basic computer operations: input/output and arithmetic/logic. Hardware elements of the computer system: data preparation devices, input devices, output devices, central processing units, auxiliary storage devices. Evolution of the electronic computer industry. Microelectronics revolution and computer industry growth. Advantages and limitations of computers. Computer types and sizes. Software considerations in computer usage. Definition and use of data. Relationships between growth of the computer industry and changes in business. Information economy: change from manufacturing to information-based economy.

2. Processing Concepts (10%) — Skill Level 2

   Data preparation and structure into logical entities of fields, records, and files. Processing cycles of input/process/output operations. Batch, transaction-oriented, and distributed processing. Computer operations: arithmetic, logic, and comparisons. Processing applications: storage and retrieval, inquiry, updating, sorting, control breaks, classifying, and summarizing. Data representation: bits, bytes, words. Binary and hex number systems. Coding systems: Hollerith, BDC, EBCDIC, ASCII. Basics of operating systems including multiprogramming, multiprocessing, buffering.

3. Input/Output (10%) — Skill Level 1

   Input/output devices: purposes, types, operations, and their relationships to the total computer system. Input/output operations, including control units, operator panels, validity switches, channels, data buffering (external and internal), and auxiliary operations. Input/output devices, including card-oriented, magnetic tape, paper tape, magnetic disk, MICR, OCR,

terminals, plotters, graphic display units, and printers. Input/output control systems, incuding I/O scheduling, blocking, error exception and correction routines, and control system advantages.

4. Memory Storage (10%) Skill Level 1

Memory storage, including functions of primary and secondary storage. Common types of primary memory, including functional definition: magnetic core, semi-conductor, recent advances such as bubble, laser, holographic. Types of memory: ROM, RAM, PROM. Common components of secondary storage, including magnetic tape and magnetic disk. File organization for tape and disk. Sequential and random access methods. Updating files using tape and disk. Virtual storage concepts.

5. Data Communications and Distributed Processing (10%) Skill Level 1

Data communications framework. Communications technology, including computer components, communications lines and network services, media speeds, modems and terminals, and controllers. Advantages and disadvantages of online, real-time, RJE, and batch transmission styles. Database goals, data types, organization, concepts and tools. Centralized vs. distributed data processing, including organizational factors. Distributed resources, including data bases, processor, support, and development. Distributed goals, including user requirements, control, simpler systems. Current applications: electronic office systems, electronic mail, automated teller machines, electronic funds transfer.

6. Computer Problem Solving (10%) Skill Level 1

Problem identification and definition. Problem analysis using structured techniques, flowcharting, and decision tables. Levels of programming available including applications programming. Solution testing, documentation and evaluation. Systems tools available to aid problem solving. Systems flowcharting. Life cycle of systems. Economic implications of solution methodology, maintenance, and design flows. Advantages and disadvantages of problem-solving techniques and tools.

7. Computer Programming (30%) Skill Level 1

Common business problem solving computer languages. Syntax of programming languages, including I/O commands, arithmetic commands, decision commands, looping commands, and other commands such as starting, stopping, or modifying. Problem solving using computer programming. Program design, coding, and keying. Compilation and debugging. Program testing and implementation. Program documentation. Programming considerations for maintenance and modification. Comparisons of data values. Conditional tests and transfers of control. Basic input/output operations. Computer arithmetic and accumulation of totals. Basic report formatting. Using prewritten software and canned programs.

8. Future of Computers in Society (10%) Skill Level 1

Trends in hardware and software technology. Trends in distributed data processing, database management systems, and administration. Networks, switching, and protocols, and office of the future. Merging of data processing, word processing and communications. Concepts of packaged software, firmware, and natural languages. Standardization in the industry. Effects of computers on organizations and individuals. Computer errors. Implications of data banks for personal and commercial privacy. Computer crime, vulnerability and security. Forms of physical and logical safeguards. Administrative controls and EDP auditing. Data processing legislation. Computer selection and contracts. Employment opportunities, including applications programmer, maintenance programmer, systems programmer, systems analyst, data base administrator, project leader, operations manager, data processing manager. Career tracks from data entry to overall management of data processing and information systems. Certification for computer professionals.

COURSE APPROACH

This is designed as the first course in computer information systems and is taught as a survey course applicable in the general business curriculum, the computer information systems major, or as a general education requirement. The teaching strategy should be directed towards increasing the computer literacy needed to function in today's technological society rather than towards technical skill development.

Emphasis throughout the course should be on problem analysis and determining how automation can contribute to satisfying business and other data processing needs. In the problem-solving methodology section of the course the problem-solving steps (definition, analysis, design, coding, testing/debugging, documentation, and evaluation) should be introduced. Each step is covered in detail through use of a well-planned and illustrated example that the instructor can develop in the classroom and expand on as the problem moves through the steps. On completion of the in-class example, one or two simple problems should be assigned to the students so they can demonstrate how the problem-solving strategies are applied.

The programming section should require problems that build upon one another. The first program should be a simple input/output program. The second will allow for arithmetic and unconditional looping. Subsequent programs will introduce program decision making and table handling capabilities. Four programs should be required to develop these programming skills. Students should be provided with sufficient equipment and software for convenient development and implementation of programs.

A research paper focusing on the technological future of computing will help tie together the final components of the course. This project will provide the student with an opportunity to explore an area of interest and provide the additional value of aiding in the development of written communication skills.

SELECTED REFERENCES

Awad, Elias M. Business Data Processing. 5th edition. Englewood Cliffs, New Jersey: Prentice-Hall, Inc., 1980.

Bohl, Marilyn. Information Processing. 3rd edition. Palo Alto, California: Science Research Associates, Inc., 1980.

Couger, J. Daniel and McFadden, Fred R. First Course in Data Processing with Basic. New York: John Wiley and Sons, 1981.

Davis, William S. Information Processing Systems. 2nd edition. Reading, Massachusetts: Addison-Wesley Publishing Co., 1981.

Dock, V. Thomas and Essick, Edward L. Principles of Business Data Processing with MIS. 4th edition. Palo Alto, California: Science Research Associates, Inc., 1980.

Edwards, Perry and Broadwell, Bruce. Data Processing: Computers in Action. Belmont, California: Wadsworth Publishing Company, 1979.

Kroenke, David M. Business Computer Systems: An Introduction. Santa Cruz, California: Mitchell Publishing, Inc., 1981.

Price, Wilson T. Introduction to Computer Data Processing. 3rd edition. New York: Holt, Rinehart and Winston, Inc., 1981.

Sanders, Donald H. Computers in Business. New York: McGraw-Hill, Inc., 1979.

Shelly, Gary B. and Cashman, Thomas J. Introduction to Computers and Data Processing. Fullerton, California: Anaheim Publishing Company, 1980.

Stern, R. A. and Stern, N. B. Principles of Data Processing. New York: John Wiley and Sons, 1979.

CIS-2 APPLICATIONS PROGRAM DEVELOPMENT I

COURSE DESCRIPTION

An introduction to computer programming in a business environment. Emphasis on the fundamentals of structured program design, development, testing, implementation, and documentation of common business-oriented applications using COBOL. Discussion and application of top-down design strategies and structured programming techniques for designing and developing problem solutions. Coverage of language syntax, data and file structures, input and output devices, and operating system facilities for implementing batch programs for report generation, input editing, table processing, and sequential file creation and access. Prerequisite: CIS-1 Introduction To Computer-Based Systems.

STUDENT OUTCOMES

To develop an understanding of and practice in the use of proper strategies and techniques for business program design and development. To develop ability to apply COBOL syntax to implement problem solutions. To gain awareness of the importance of correct programming methodology and to develop motivation for the use of proper programming techniques. To become aware of common business programming applications. To gain the background for further study of software design and computer programming in a business environment.

COURSE CONTENT

1. Structured Design Concepts (5%) — Skill Level 1

Structured and algorithmic problem-solving. Problem analysis and functional decomposition. Correlations between problem structure and program structure. Module definitions. Relationships between program modules. Data flow approaches to design. Data structure approaches to design. Contrasts with process flow approaches to design. The need for systematic methodologies for program development. Graphical tools for structured design.

2. Structured Programming Concepts (10%) — Skill Level 2

Methods of specifying program logic. Basic program control structures--sequences, decisions, and iterations. Pseudocode, IPO charts, decision tables, decision trees, flowcharts, and other verbal and graphical specification tools. Uses and limitations of flowcharts.

3. The Programming Process (10%) — Skill Level 2

Program specifications review. Program design. Specificaiton of module logic. Programming. Program coding and keying. Program compilation, testing, and implementation. Documentation and programming style. Programming considerations for maintenance and modifications. Peer reviews during program development. Programming in a batch environment. Operating system facilities for program development and execution.

4. File Processing (15%) Skill Level 2

Creation and use of sequential files. Sequence checking. Editing and validation of input. The concept of a key in sequential organization. Relationship between sequence of input and sequence of output. The language interface for using sequential files as input and creating sequential files as output. Logical versus physical input/output. Volumes, files, records, and fields. Internal and external representation of data in primary storage and on secondary storage devices. Language interface to various devices using operating systems facilities. Controlling accuracy and creating audit trails. Exception reporting.

5. Program Decision Making (15%) Skill Level 2

Comparison of data values and data types. Conditional tests and transfers of control. Single-level decision making. Multiple-level, nested tests and implementation of the case construct. Decision tables and decision trees for analysis of decision problems. Multiple-page report formatting. Language syntax for logical operations.

6. Control Break Processing (15%) Skill Level 2

Control break concepts. Multiple-level control breaks. Group indication and report formatting. Accumulation of subtotals and selective printing of subtotals. Techniques for "LOOK-AHEAD" reading of files. Design strategies and program structures for control break processing. File structures to support multiple-level processing and reporting.

7. Table Processing (15%) Skill Level 2

Single and multi-dimensional tables. Defining and loading tables at load time and execution time. Loading, updating, and storing tables on secondary storage devices. Table searching using sequential and binary techniques. Use of subscripts and indexes. Language syntax associated with setting up and handling tables.

8. File Creation and Processing (15%) Skill Level 2

Organization of sequential files. The concept of record keys. Magnetic tape and sequential disk files. File creation and processing. Data editing and validation techniques. Sequence checking. Batch controls. Exception reporting. Language syntax for sequential file creation and processing. Operating system facilities for file processing.

COURSE APPROACH

This is designed as a first programming course. As such, it will provide the fundamental problem-solving approaches that will be expanded upon throughout the other courses in the curriculum. Although learning a higher-level language syntax will be a necessary outcome of this course, the primary benefits will be in the problem-solving strategies, programming techniques, and business data processing concepts that are learned. The position of the program in the life-cycle of a system should be stressed. The vehicle language for this course should be COBOL.

Emphasis throughout the course should be on program analysis, top-down design, functional decomposition, and module design based on the three primary program control structures--sequences, decisions, and iterations. The particular design strategy should be representative of the generally popular data flow or data structures approaches. An effective method of documenting programs should be adopted and used throughout this and subsequent courses.

Programming projects should illustarte:

1. Basic input/output operations.
2. Arithmetic operations and report formatting.
3. Accumulation of final totals and advanced report editing.
4. Program decision making and multiple-page reports.
5. Control break processing.
6. Table processing.
7. Sequential file creation, access, and reporting including data editing and validation.

Most programs will be developed and implemented individually in batch mode, however, emphasis should be given to peer reviews of program design, programming, and coding. On-line program development techniques should be included when resources are available.

SELECTED REFERENCES

Bohl, Marilyn. Tools for Structured Design. Palo Alto, California: Science Research Associates, Inc., 1978.

De Marco, Thomas. Structured Analysis and Systems Specification. Englewood Cliffs, New Jersey: Prentice-Hall, Inc., 1979.

Feingold, Carl. Fundamentals of Structured COBOL Programming. Dubuque, Iowa: Wm. C. Brown, 1973.

Gane, Chris and Sarson, Trish. Structured Systems Analysis. Englewood Cliffs, New Jersey: Prentice-Hall, Inc., 1979.

Grauer, Robert T. A COBOL Book of Practice and Reference. Englewood Cliffs, New Jersey: Prentice-Hall, Inc., 1981.

Grauer, Robert T. COBOL--A Vehicle for Information Systems. Englewood Cliffs, New Jersey: Prentice-Hall, Inc., 1981.

Grauer, Robert T., and Crawford, Marshal A. Structured COBOL--A Pragmatic Approach. Englewood Cliffs, New Jersey: Prentice-Hall, Inc., 1981.

Jackson, Michael A. Principles of Program Design. New York: Academic Press, 1975.

McCracken, Daniel D. A Simplified Guide to Structured COBOL Programming. New York: John Wiley and Sons, 1976.

Murach, Michael and Noll, Paul. Structured ANS COBOL Part I. Fresno, California: Murach and Associates, 1979.

Murach, Michael and Noll, Paul. Structured ANS COBOL Part II. Fresno, California: Murach and Associates, 1979.

Noll, Paul. The Structured Programming Cookbook. Fresno, California: Murach and Associates, 1978.

Orr, Kenneth. Structured Systems Development. New York: Yourdon Press, 1977.

Philippakis, Andreas S. and Kazmier, Leonard J. Information Systems Through COBOL. 2nd edition. New York: McGraw-Hill, 1977.

Shelly, Gary B., and Cashman, Thomas J. Advanced Structured COBOL. Fullerton, California: Anaheim Publishing, 1978.

Shelly, Gary B., and Cashman, Thomas J. Introduction to Computer Programming-Structured COBOL. Fullerton, California: Anaheim Publishing, 1977.

Yourdon, Edward, Gane, Chris, Sarson, Trish and Lister, Timothy R. Learning to Program in Structured COBOL. Englewood Cliffs, New Jersey: Prentice-Hall, Inc., 1979.

Yourdon, Edward and Constantine, Larry L. Structured Design. Englewood Cliffs, New Jersey: Prentice-Hall, Inc., 1979.

Yourdon, Edward. Techniques of Program Structure and Design. Englewood Cliffs, New Jersey: Prentice-Hall, Inc., 1975.

CIS-3 APPLICATIONS PROGRAM DEVELOPMENT II

COURSE DESCRIPTION

A continuation of CIS-2. Emphasis on structured methodology of program design, development, testing, implementation, and documentation of common business-oriented applications using COBOL. Includes coverage of sequential and random access files and processing techniques and development of programs and systems of programs for batch and interactive environments. Prerequisite: CIS-2, Applications Program Development I.

STUDENT OUTCOMES

To develop ability to design and implement in COBOL programs and systems of programs that utilize advanced features of the language. To develop an understanding of common business-oriented applicatons and capabilities of the language in both batch and interactive processing environments. To develop an understanding of and ability to apply structured program design, programming, and implementation methodologies. To develop motivation for applying proper software development techniques.

COURSE CONTENT

1. Programs And Systems Of Programs (10%) — Skill Level 1

Design and graphic representation of programs and systems of programs. Interprogram communication and intermediate files. Packaging of software into program modules. Development of menu-driven systems of programs and subprograms.

2. Program Design (10%) — Skill Level 2

Concept of top-down program development and functional decomposition. Data flow and data structure approaches to design. Graphical tools for representing program structures. Concepts of module coupling and cohesion. Utilizing paragraphs, sections, subprograms, and programs for module implementation. General structural modules for common business applications. Coordination and communications in modular systems.

3. Module Design (15%) — Skill Level 2

Development of detailed module logic. Use of sequences, decisions, and iterations control structures. Use of pseudocode and flowcharts to represent structured logic. Use of decision tables and decision trees to express complex logic. Implications of module design on program testing. Algorithm development for sorting and searching. Module specifications and documentation.

4. Programming In COBOL (25%) — Skill Level 3

Implementing designs in COBOL. Language conventions for implementing control structures. Converting Pseudocode and flowcharts to COBOL syntax. Software development tools: Source libraries, data dictionaries, compiler options, and debugging aids.

5. COBOL File Processing (20%) Skill Level 2

Nature of sequential, direct, and indexed file organization. Sequential and random access. COBOL language syntax and coding conventions for file processing. File organization considerations. Systems utilities for file processing. Language interface with physical files.

6. Advanced Language Features (20%) Skill Level 2

Use of sort facility with using and giving options. Use of sort input and output procedures. Comparisons of COBOL and utility sorts. String manipulation using the concepts of string and unstring. Writing conversational and menu - driven programs. Programming in a communications environment. Use of the report writer feature.

COURSE APPROACH

This is designed as a second programming course emphasizing advanced program design and implementation of advanced language features. The main orientation of the course is to file maintenance activities and the development of programs and systems of programs to carry them out.

A system of programs that will emphasize the flow of data and the relationship of one program to another is recommended for the laboratory activities. The student activities should include the design and coding of programs, subroutines, and subprograms. While the programs are to be individually developed, the team design concept should be taught.

The teaching methodology for this course should include lecture, discussion, and illustration and clarification of concepts. The instructor should provide the information system design and the student will be expected to develop the specific design of the programs. Emphasis should be given to peer review of program design, systems design and coding. Class time should be devoted to these structured walkthroughs.

Included in the exercises should be requirements to design, program, code, and test realistic business-type programs that illustrate the following applications using a system concept:

A. Input/output operations.
B. Arithmetic operations and basic report formatting.
C. Accumulation of final totals and advanced report editing.
D. Program decision making and multiple-page reports.
E. Control breaks.
F. Table processing methods.
G. Sequential file creation, access and reporting including basic data editing.
H. Transfer of information from one program to another.
I. Data edit/validation.
J. Updating sequential, indexed, and direct files.
K. Extract and reporting.

L. Interactive and conversational programs.
M. Sorting and merging files.
N. Random access method, space computation, space allocation, file building and rebuilding, and file backups and reloads.
O. System flowcharts.
P. Cost analysis and expected savings related to programming efficiencies.
Q. Program libraries concept.
R. Structured walkthroughs.
S. Use of programming standards.

SELECTED REFERENCES

Bohl, Marilyn. Tools for Structured Design. Palo Alto, California: Science Research Associates, Inc., 1978.

De Marco, Thomas. Structured Analysis and Systems Specification. Englewood Cliffs, New Jersey: Prentice-Hall, Inc., 1979.

Feingold, Carl. Fundamentals of Structured COBOL Programming. Dubuque, Iowa: Wm. C. Brown, 1973.

Gane, Chris and Sarson, Trish. Structured Systems Analysis. Englewood Cliffs, New Jersey: Prentice-Hall, Inc., 1979.

Grauer, Robert T. A COBOL Book of Practice and Reference. Englewood Cliffs, New Jersey: Prentice-Hall, Inc., 1981.

Grauer, Robert T. COBOL--A Vehicle for Information Systems. Englewood Cliffs, New Jersey: Prentice-Hall, Inc., 1981.

Grauer, Robert T. and Crawford, Marshal A. Structured COBOL--A Pragmatic Approach. Englewood Cliffs, New Jersey: Prentice-Hall, Inc., 1981.

Jackson, Michael A. Principles of Program Design. New York: Academic Press, 1975.

McCracken, Daniel D. A Simplified Guide to Structured COBOL Programming. New York: John Wiley and Sons, 1976.

Murach, Michael and Noll, Paul. Structured ANS COBOL Part I. Fresno, California: Murach and Associates, 1979.

Murach, Michael and Noll, Paul. Structured ANS COBOL Part II. Fresno, California: Murach and Associates, 1979.

Noll, Paul. The Structured Programming Cookbook. Fresno, California: Murach and Associates, 1978.

Orr, Kenneth. Structured Systems Development. New York: Yourdon Press, 1977.

Philippakis, Andreas S. and Kaznier, Leonard J. Information Systems Through COBOL. 2nd edition. New York: McGraw-Hill, 1977.

Shelly, Gary B. and Cashman, Thomas J. Advanced Structured COBOL. Fullerton, California: Anaheim Publishing, 1978.

Shelly, Gary B. and Cashman, Thomas J. Introduction to Computer Programming-Structured COBOL. Fullerton, California: Anaheim Publishing, 1977.

Yourdon, Edward, Gane, Chris, Sarson, Trish and Lister, Timothy R. Learning to Program in Structured COBOL. Englewood Cliffs, New Jersey: Prentice-Hall, Inc., 1979.

Yourdon, Edward and Constantine, Larry L. Structured Design. Englewood Cliffs, New Jersey: Prentice-Hall, Inc., 1979.

Yourdon, Edward. Techniques of Program Structure and Design. Englewood Cliffs, New Jersey: Prentice-Hall, Inc., 1975.

CIS-4 SYSTEMS ANALYSIS METHODS

COURSE DESCRIPTION

Overview of the system development life cycle. Emphasis on current system documentation through the use of both classical and structured tools/techniques for describing process flows, data flows, data structures, file designs, input and output designs and program specifications. Discussion of the information gathering and reporting activities and of the transiton from analysis to design. Prerequisite: CIS-2, Applications Program Development I.

STUDENT OUTCOMES

To develop a basic understanding of information systems, general system solutions and the discipline of systems analysis in relation to the information system life cycle. To acquire skill in applying the tools, techniques and concepts of both classical and structured analysis to information systems development.

COURSE CONTENT

1. System Development Life Cycle Overview (10%) Skill Level 1
   Purpose of system development life cycle and fundamental concepts. Introduction to major phases: investigation and feasibility study, analysis and general design, detailed design and implementation. Elements of an information system: hardware, software, procedures and human resources.

2. Classical Documentation Tools and Techniques (20%) Skill Level 2
   Process flow approaches to system documentation. System flowcharting. File and record design considerations. Input/output documentation. Input, output and file media devices and organization. Program specification techniques. Design tradeoffs. Overview of hardware and software configurations.

3. Structured Documentation Tools and Techniques (25%) Skill Level 2
   Elements of data flow diagrams: processes, data flows and data stores. Process descriptions. Use of structured English, decision tables and decision trees to express processing logic. Data descriptions. Concepts of data dictionary and data structure diagrams.

4. Documenting the Current Physical System (15%) Skill Level 2
   Output specifications. Documentation of physical output formats. Input design considerations. File layout and design considerations. System flowcharting. Program specifications. Data flow approaches to system documentation. Data flow diagramming of current physical system. Data dictionary creation.

5. Derivation of Current Logical System (10%) Skill Level 1
Concepts of abstraction and hierarchical decomposition. Converting physical data flows and processes into logical equivalents. Logicalization of physical data dictionary specifications. Derivation of logical process specifications. Transitions to systems analysis and design.

6. Information Gathering and Reporting (20%) Skill Level 2
Sources of information. Interviewing and survey techniques. Oral and written communications. Report formats and presentations. Design of user documentation. Human factors affecting analysis and documentation.

COURSE APPROACH

This is designed as the first course in systems development. On the one hand it is designed to stand alone as an introduction to the system development life cycle in the recommended community college subset of the curriculum; on the other hand, it is designed as the first course in the three-course systems sequence in the four-year version of the curriculum. Its primary emphasis is on the methods, tools, and products of systems analysis geared toward system documentation rather than toward the formal strategies and techniques of systems analysis and systems design. Analysis tools are employed to document an existing system from both a physical and logical perspective. Discussions of classical and structured methodologies are interwoven as a foundation for advanced study of analysis and design.

The major components of the system development life cycle and their interrelationships as well as the role of a system development methodology in achieving control of the development effort should be well understood. The analysis phase will be stressed most heavily, but the concerns and products of the design phase should also be briefly introduced. Several well developed examples would greatly help the process. Community colleges can tailor these topics to suit local needs and increase the ability of the course to stand alone.

Considerable emphasis should be placed on learning to use the tools of structured analysis. These tools can be viewed first as useful documentation and communication tools by employing them to document parts of an existing system. Their use as general design tools can be introduced here by briefly covering the structured analysis process.

Rather than employing a major systems project, this course will rely on short exercises to illustrate the key concepts and develop skills in using the various tools. The success of the course will be greatly enhanced by having a good collection of examples and sample products.

The approach is to use individual projects and exercises, as opposed to a team analysis/design project. The emphasis is on mastering the tools and techniques of both classical and structured systems analysis. This could be accomplished via individual exercises or a comprehensive project to document an existing system supplied by the instructor.

Throughout the course there must be a heavy emphasis on:
- problem solving skills and
- communication skills as employed by the analyst.

SELECTED REFERENCES

Biggs, Charles L., Birks, Evan G., and Atkins, William. Managing the Systems Development Process. Englewood Cliffs, New Jersey: Prentice-Hall, Inc., 1980.

DeMarco, Thomas. Structured Analysis and System Specification. Englewood Cliffs, New Jersey: Prentice-Hall, Inc., 1979.

Gane, Chris and Sarson, Trish. Structured Systems Analysis. Englewood Cliffs, New Jersey: Prentice-Hall, Inc., 1979.

Leeson, Marjorie. Systems Analysis and Design. Palo Alto, California: Science Research Associates, Inc., 1981.

Shelly, Gary B. and Cashman, Thomas J. Business Systems Analysis and Design. Fullerton, California: Anaheim Publishing Co., 1975.

Weinberg, Victor. Structured Analysis. Englewood Cliffs, New Jersey: Prentice-Hall, Inc., 1980.

CIS-5 STRUCTURED SYSTEMS ANALYSIS AND DESIGN

COURSE DESCRIPTION

Advanced study of structured systems development. Emphasis on strategies and techniques of structured analysis and structured design for producing logical methodologies for dealing with complexity in the development of information systems. Prerequisites: CIS-3, Applications Program Development II and CIS-4, Systems Analysis Methods.

STUDENT OUTCOMES

To understand the role of systems analysis and design within the systems development life cycle. To develop the ability to apply structured strategies and techniques to system analysis and system design phases. To develop the ability to produce system specifications and implementation and test plans for developing and implementing information systems that satisfy user requirements.

COURSE CONTENT

1. The System Development Life Cycle (5%) — Skill Level 2
   Review of the system development life cycle. The role of structured analysis and structured design within the framework of the system life cycle. Products of analysis and design. Review of investigation and feasibility phases. Definition of user requirements.

2. Documenting the Current Physical System (10%) — Skill Level 3
   Information gathering sources and techniques. Creation of physical data flow diagrams. Use of leveled data flow diagrams. Contrasts with process flow approaches to analysis and documentation. The concepts of abstraction and hierarchical problem solving applied to the system development process.

3. Derivation of the Current Logical System (15%) — Skill Level 2
   Transformation of physical into logical data flow diagrams. Development of a data dictionary. Leveled data flow diagrams and derivation of process descriptions. Use of structured English, decision tables, and decision trees. Control structures for expressing processing logic.

4. Modeling the New Logical System (20%) — Skill Level 2
   The structured specification. Derivation of data flow diagrams for new logical systems. Derivation of process descriptions. Derivation of logical data structures through normalization of file structures. Packaging the structured specification. Operational, economic, psychological and political considerations of the new system.

5. Derivation of the New Physical System (20%) Skill Level 2
Establishing human-machine interfaces and boundaries. Transition to design. Derivation of system architecture. General design strategies: transform analysis, transaction analysis and data structure approaches to design. Preliminary packaging considerations: structure of system into programs and subprograms.

6. Detailed Design (20%) Skill Level 2
Factoring techniques for designing system branches. Design criteria: coupling, cohesion and distribution of control. Module design and specification. Packaging and optimization of designs. System controls. Data structure approaches to design.

7. Implementation Considerations (10%) Skill Level 2
Derivation of test plan and test data. Implementation strategies. System testing. Acceptance testing. Evaluation considerations.

PROJECTS AND EXERCISES

Ideally this course will employ a semester long project to do at least major portions of the analysis and design of an information system in a case study environment. The project would begin with the results of a Feasibility Study (or later if desired: for example, with a physical model of a current system). It would end with the physical packaging of the new system, the detailed design at a structure chart level of at least some of the processing modules, a test plan for these modules, and an implementation plan. (NOTE: The topics in the outline parallel the steps involved in the project. The time percentages listed on the outline refer to class time, NOT to project time.)

Individual problem sets could be used in place of a single case study project. They should require the student to develop a set of logical system specifications (data flow diagrams, data dictionary, process specifications and data structure diagrams), physical system designs (hierarchy charts and module specifications), and an implementation plan for a system of moderate complexity.

COURSE APPROACH

This course covers the methodology of structured systems development concentrating on the analysis and design phases of the system development life cycle. It is a course in problem solving wherein the student learns and applies strategies and techniques for dealing with complexity. Both data flow and data structure approaches to design should be investigated.

It is strongly recommended that the central student activity of the course be a semester long case study type project to do major portions of the analysis and design of a (simplified) medium sized information system. The project should concentrate on the structured analysis process, the transition from analysis to design, the characteristics of good design and finally on the techniques of structured design. Student understanding will be enhanced if the analysis and design activities can be placed in the context of some specific system development methodology. While all methodologies vary slightly, there is a vast difference in understanding gained by living within and dealing with the rigor of a specific system development methodology as opposed to reading about methodologies in general. The topics in the course outline parallel the steps involved in the project. Note that the time percentages listed on the outline refer to class time, NOT to project time.

Throughout the course there must be a heavy emphasis on:

- problem solving skills and
- communication skills.

Students should engage in the process of analysis and design instead of merely concentrating on the products.

SELECTED REFERENCES

Biggs, Charles L., Birks, Evan G., and Atkins, William. Managing the Systems Development Process. Englewood Cliffs, New Jersey: Prentice-Hall, Inc., 1980.

Gane, Chris and Sarson, Trish. Structured Systems Analysis. Englewood Cliffs, New Jersey: Prentice-Hall, Inc., 1979.

Jackson, Michael A. Structured Systems Analysis. New York: Academic Press, 1975.

Myers, Glenford. Composite/Structured Design. New York: Van Nostrand Reinhold, 1978.

Orr, Kenneth. Structured Systems Design Methodolgy. Topeka, Kansas: Ken Orr and Assoc., 1980.

Page-Jones, Meilir. The Practical Guide to Structured Systems Design. New York: Yourdon Press, 1980.

Weinberg, Victor. Structured Analysis. Englewood Cliffs, New Jersey: Prentice-Hall, Inc., 1979.

Yourdon, Edward and Constantine, Larry L. Structured Design. Englewood Cliffs, New Jersey: Prentice-Hall, Inc., 1979.

See also the January issue (each year) of Computing Newsletter, published by CYSYS, Box 7345, Colorado Springs, Colorado 80933.

CIS-6 DATABASE PROGRAM DEVELOPMENT

COURSE DESCRIPTION

Introduction to application program development in a database environment with an emphasis on loading, modifying and querying the database using a host language (COBOL). Discussion and application of data structures, indexed and direct file organizations, models of data including hierarchical, network and relational. Discussion of storage devices, data administration and data analysis, design and implementation. Prerequisite: CIS-5, Structured Systems Analysis and Design.

STUDENT OUTCOMES

To gain in-depth understanding of the development of application programs in a database environment. To learn and apply data structures for the physical implementation of storage of data. To understand, from the perspective of an applications programmer, storage device characteristics and associated physical input/output strategies. To become aware of different approaches to data modeling. To understand the different roles in data administration and the bases for evaluating issues surrounding data administration. To gain practice in applying techniques of data organization in a database environment employing the host language facility of a data base management system.

COURSE CONTENT

1. Overview (10%) — Skill Level 2

   Data and reality. Concept of an information space - entities, attributes and values. Multiple views of data - end-user, application programmer, database technician and database administrator. Database development and the system life-cycle. Database issues - security, privacy, integrity, data independence and advantages/ disadvantages over traditional file environment.

2. Storage Device Characteristics and Physical Input/Output (5%) — Skill Level 3

   Sequential files and media - blocking, spanned records and variable length records. Random access files and media - volume, cylinder, track, sector, blocking, spanned records, spanned files and variable length records.

3. Indexed Organized Files (10%) — Skill Level 3

   Full index, block index, hierarchical index and multiple key indicies. Sequential, random and dynamic access. Overflow.

4. Direct File Organization (10%) — Skill Level 3

   Bucketing. Relative addressing. Hash addressing. Packing density. Overflow.

5. Applied Data Structures (10%) Skill Level 2

Linked list structures, pointer chains and rings. Single (one-way), double (bi-directional) and multiple (inverted) linking strategies. Insertion, deletion and traversal of chains. Trees - balanced, unbalanced, binary and pre-order traversal. Networks. Data structure diagram - 1:1, 1:N, M:N relationships, cycles and loops.

6. Data Model Overview (5%) Skill Level 2

What is a data model? Data definition language, DDL. Data manipulation language, DML. Representations of entities, attributes and relationships. Three prominent data models - hierarchical, network, and relational. Framework for evaluation of data models.

7. Hierarchical Data Model (10%) Skill Level 2

DDL terminology and concepts - field, segment, database record and database. DML terminology and concepts - Get Uniques, Get Next, Get Next within Parent, Get Hold Unique, Get Hold Next and Get Hold Next within Parent. Replace, delete and insert. Physical database record and logical database record. Sample system, IBM's Information Management System. Advantages/disadvantages of the hierarchical approach.

8. Network Data Model (15%) Skill Level 2

DDL terminology and conepts - data item, vector, repeating group, record, set (member and owner), realm and area, sub-schema and schema. DML terminology and concepts - Ready, Find, Get, Modify, Erase, Connect, Disconnect. Relationship implementation. Sample system - Cullinane's IDMS. Advantages/disadvantages of the network approach.

9. Relational Data Model (15%) Skill Level 2

DDL terminology and concepts - relation, attributes, attribute name, domain, tuple, primary key, candidate key, concatenated key and normalization. DML terminology and concepts - Projection, Join, Division, Relational Algebra and Relational Calculus. Sample system - IBM System R and SEQUEL. Advantages/disadvantages of relational approach.

10. Database Administration (10%) Skill Level 2

Roles - Database Administrator, Database Technician, Application Programmer, Systems Analyst/Designer. Security, privacy, integrity, backup and recovery.

COURSE APPROACH

This course is designed as a first course in database. However, the student will have had, as prerequisites, at least two courses in programming, a course in systems analysis and a course in systems design. Therefore, the student will be familiar with the program development process, have a sound background in a higher-level language, COBOL, and an appreciation for random access file organizations both from an applications programming perspective and from a systems analysis/design perspective.

The primary benefit the student should gain from this course is an in-depth appreciation of modeling complex organizations of data that support multiple differing user views with the aid of a DBMS. It should be noted that this emphasis is on use of the DBMS not how to design/develop one.

The subject is introduced by evaluating the evolution of applications sharing data leading to the need for DBMS's. This is supported with a conceptual model of an organization's information needs represented spacially with dimensions of entities, attributes and values. Environmental and social impacts are also discussed.

The foundation for investigation of various strategies of DBMS implementation is built from the bottom up. Starting with storage device characteristics with associated physical input/output routines, coupled with access methods that make use of these to provide for a realization of logical data structure needs while accomodating physical device constraints.

Ideally the student should have an opportunity to complete programming exercises with each of the three data models. However, due to time constraints and the possible lack of resources, it is anticipated that some of the exercises may only be completed on paper. Note that there are several "DBMS" avaliable specifically tailored for the student environment.

The discussion of database administration and data analysis/design is intended to be introductory in nature providing a motivation for possible advanced study in database.

The programming applications for this course include; (1) compilation of a schema (supplied by instructor), (2) compilation of a sub-schema derived from the schema based upon a user view (again, supplied by instructor), (3) use of DML verbs through a high-level language (COBOL) to maintain (load, modify, delete and query) the database described in 1 and 2, and (4) optional use of a query language facility to access the database.

SELECTED REFERENCES

Atre, Steve. Data Base: Structured Techniques for Design, Performance and Management. New York: John Wiley and Sons, Inc., 1980.

Borkin, Sheldon A. Data Models: A Semantic Approach for Database Systems. Cambridge, Massachusetts: MIT Press, 1980.

Flavin, Michael. Fundamental Concepts of Information Modeling. New York: Yourdon Press, 1981.

Hutt, A. T. A Relational Database Management System. New York: John Wiley and Sons, Inc., 1979.

Johnson, Leroy F. and Cooper, Rodney H. File Techniques for Database Organization in COBOL. Englewood Cliffs, New Jersey: Prentice-Hall, Inc., 1981.

Katzan, Harry. Computer Data Management and Data Base Technology. New York: Van Nostrand Reinhold, 1975.

Kroenke, David. Database Processing: Fundamentals, Modeling, Applications. Palo Alto, California: Science Research Associates, Inc., 1977.

Lyon, John K. The Database Administrator. New York: John Wiley and Sons, Inc., 1976.

Martin, James. Computer Database Organization. 2nd edition. Englewood Cliffs, New Jersey: Prentice-Hall, Inc., 1977.

Meldman, Monte J., McLeod, Dennis J., Pellicore, Robert J., and Squire, Morris. RISS: A Relational Data Base Management System for Minicomputers. New York: Van Nostrand Reinhold, 1978.

Olle, T. William. The CODASYL Approach to Data Base Management. New York: John Wiley and Sons, Inc., 1978.

Tsichritzis, Dennis C. and Lochovsky, Frederick H. Data Base Management Systems. New York: Academic Press, 1977.

Ullman, Jeffrey D. Principles of Database Systems. Rockville, Maryland: Computer Science Press, 1980.

CIS-7 APPLIED SOFTWARE DEVELOPMENT PROJECT

COURSE DESCRIPTION

Application of computer programming and system development concepts, principles and practices to a comprehensive system development project. A team approach is used to analyze, design and document realistic systems of moderate complexity. Use of project management methods, project scheduling and control techniques, formal presentations and group dynamics in the solution of information systems problems. Development of a database to support the system. Prerequisites: CIS-5, Structured Systems Analysis and Design and CIS-6, Data Base Program Development.

STUDENT OUTCOMES

To develop the ability to synthesize and integrate systems analysis, systems design, programming and business and information systems concepts, principles and practices in the development of computer-based information systems. To develop the ability to apply technical, managerial, communications and interpersonal skills to a realistic, moderately complex system development project set in a team environment.

COURSE CONTENT

1. Project Management Concepts (20%) — Skill Level 3
   Project planning and task definition. Scheduling work activities and tasks. Scheduling and control techniques: GANTT, PERT, CPM. Project resource budgeting and cost control. Selection of project team personnel. Organization/team structure. Team/group dynamics and conflict management. Systems change impacts.

2. Analysis of Current System (20%) — Skill Level 3
   Feasibility study and project definition. Project planning and organization. Data collection: interviews, observation and documentation. Current system description: flowcharts, HIPO and narrative. Identification of systems problems: bottlenecks, duplication and complexity. Identification of systems requirements: hardware, software, economics, personnel, environment and government.

3. Design of Proposed/Modified System (20%) — Skill Level 3
   Identification of current and future objectives. Priority setting among system's requirements and objectives. Building top-down flow-charts: systems level, applications level, sub-applications level and program level. Designing database specifications. Designing input and output specifications: screens, menus and reports. Designing program specifications.

4. Implementation (25%) — Skill Level 3
   Selecting appropriate language. Building top-down program modules. Building top-down program test-stubs. Testing modules in a a top-down fashion. Conducting structured walkthroughs. Building test data sets.

Testing programs and systems. Systems performance evaluation. Project performance evaluation.

5. Communications (15%) Skill Level 3
User interviews. Writing systems and program documentation. Writing user manuals. Writing operations manuals. Interim and final project reports. Formal presentations of project results.

COURSE APPROACH

This course is a practicum in systems development utilizing real projects or contrived cases that require application of the activities of the system development life cycle. The projects are set in a team environment with the supervising faculty member serving as consultant and evaluator. Students are required to apply concepts, principles and problem-solving strategies, techniques and tools learned in previous CIS courses as well as knowledge and abilities acquired in other business disciplines to the case problems.

CIS-8 SOFTWARE AND HARDWARE CONCEPTS

COURSE DESCRIPTION

A survey of technical topics related to computer systems with emphasis on the relationships between hardware architecture, system software, and applications software. The architecture of processors and storage systems are explored and the implications for systems software design are covered along with the impact of hardware and system software design on the development of application programs in a business environment. Prerequisite: CIS-3, Applications Program Development II.

STUDENT OUTCOMES

To develop a basic understanding of hardware and system software architecture applicable to the development of application software. To develop a broad background in technical topics of computer systems so that information systems specialists can converse with systems support personnel and can maintain a knowledge of trends and developments in computer systems.

COURSE CONTENT

1. Computer System Components (10%) — Skill Level 1
   Overview of simple CPU, main storage and I/O processor relationships. CPU operating cycle with dicussion of cycle timing and the role of registers. Program counter, status register, instruction register and accumulator vs. general purpose registers. Main storage operations through data and address registers. I/O processor relationships with CPU and main storage.

2. Main Storage Organization (10%) — Skill Level 2
   Bits, bytes and word concepts. Absolute addressing, relative addressing and indexing. Need for program/data relocation and the impact on addressing for this process. The use of indexing in table handling. Levels of main storage (cache, etc.), stacks and bus architecture.

3. Instruction Sets and Data Representation (20%) — Skill Level 2
   Binary, octal and hexadecimal systems. Numeric data types, their representations and their use in HOL's (i.e., integer, COMP-3, etc.). Character data (ASCII and EBCDIC) and its use in HOL's (e.g., DISPLAY). Instruction set for CPU and the relationship to addressing and data types. Instruction set for I/O processor.

4. Program Translation Concepts (20%) — Skill Level 2
   Assembler language concepts (instruction formats, mnemonics, symbolic addressing, pseudo-operations and macros). The assembly process (labels, location counter, external references and arguments). Operation of linking and loading. Compilation of HOL statements (e.g., MOVE, COMPUTE, READ). Subroutine and copy libraries. Dump analysis.

5. Operating Systems Concepts (20%) Skill Level 2
Role of operating systems in resource management. Scheduling, data management (including I/O interrupts) and memory management (including virtual memory concepts). Relationship of application program design to memory management, scheduling and I/O processing.

6. Secondary Storage Concepts (10%) Skill Level 2
Characteristics of disks, tapes and mass storage units. Concepts of tracks, sectors, cylinders and blocks. Access time. Roles of channels and other types of I/O processors. File design and space allocation.

7. Advanced Concepts (10%) Skill Level 1
Multi-processor systems, networks, front-end processors and database processors. Microprogramming. Array processors.

COURSE APPROACH

This course is designed as a survey course covering technical hardware and software concepts and terminology. Emphasis should be on broad, general principles applicable to most computer systems. The common features of microcomputers, minicomputers, and mainframe computers should be examined. Technical concepts should be related to the problem-solving needs of the computer information systems specialist. The student is assumed to have programmed in a higher-level language (COBOL) and be familiar with the concepts of programs, subroutines and data storage. Although not a programming course, programming assignments should be used to illustrate such concepts as data representations, addressing modes, subroutines with arguments, program and copy libraries, dump analysis and file concepts.

SELECTED REFERENCES

Books:

Barron, D. W. Assemblers and Loaders. New York: North-Holland, 1978.

Eadie, Donald. Minicomputers: Theory and Operation. Reston, Virginia: Reston Publishing Co., Inc., 1979.

Freeman, Peter. Software Systems Principles. Palo Alto, California: Science Research Associates, Inc., 1975.

Gear, C. William. Computer Organization and Programming. New York: McGraw-Hill, 1980.

Madnick, Stuart and Donovan, John. Operating Systems. New York: McGraw-Hill, 1974.

Stone, Harold. Introduction to Computer Architecture. 2nd edition. Palo Alto, California: Science Research Associates, Inc., 1980.

Periodicals:

"A Primer on Virtual Memory," Mini-Micro Systems, (March 1979) pp. 39-45.

Bulman, David M. "Stack Computers: An Introduction," Computer, Volume 10, No. 5 (May 1977) pp. 14-28.

El-Ayat, Khaled A. "The INTEL 8089: An Integrated I/O Processor," Computer, Volume 12, No. 6 (June 1979) pp. 67-78.

Hoagland, Albert S. "Storage Technology: Capabilities and Limitations," Computer, Volume 12, No. 5 (May 1979) pp. 12-19.

Kerr, Douglas S. "Database Machines with Large Content Addressable Blocks and Structural Information Processors," Computer, Volume 12, No. 3 (March 1979), pp. 64-79.

"Microcomputer Operating System Comes of Age," Mini-Micro Systems, (October 1980) pp. 97-119.

Reddi, S. S. and Peastel, E. A. "A Conceptual Framework for Computer Architecture," Computing Surveys, (June 1976) pp. 277-300.

Tjaden, Gary and Cohn, Martin. "Some Considerations in the Design of Mainframe Processors with Microprocessor Technology," Computer, Volume 12, No. 8 (August 1979) pp. 68-74.

CIS-9 OFFICE AUTOMATION

COURSE DESCRIPTION

Office information and decision support systems are examined as emerging and critical elements in business data and information systems. Emphasis is given to information processing considerations at the systems level, including analysis and management of support activities such as data and records management, electronic filing and retrieving systems, word processing, micro and reprographics, and (tele-) communications. The course includes discussion of person/machine interfaces and appraisals of current and future technological trends and their impacts on data processing and on the office environment. Prerequisite: CIS-4, Systems Analysis Methods.

STUDENT OUTCOMES

To increase understanding of business support systems--their designs, personnel, organization and administration. To improve ability to render decisions leading to design of user-oriented, integrated business and office information systems which are both cost-effective and productive. To gain background for further study in the area of information management and design of effective support systems for decision making.

COURSE CONTENT

1. Organizational Concepts (15%) — Skill Level 1

   Review of primary systems such as production, finance and marketing and their relationships with support operations. Organizational patterns for managing supporting systems. Dispersed versus centralized location and control. Manual versus automated and electronic applications. Organizing principles and problems. Coordination and political issues. Work flow/information flow. Corporate decision hierarchies. Top-down versus bottom-up design approaches. Management and reporting by exception.

2. Office: Traditional and Emerging (10%) — Skill Level 1

   Historical perspectives. Changes in coordination of OIS and managerial policy. Manual-mechanical and automated systems. Impact of electronic data processing, word processing and related systems. Productivity issues. Privacy issues. Non-technical nature of the office user.

3. Characteristics of Major Support Systems (30%) — Skill Level 2

   Data and records management. Word processing. Micro and reprographics. Facsimile. COM/CIM/CAR. Communications: voice, data and video. Executive support systems, including calendar management, scheduling, personal computing, electronic mail, data retrieval and review and video conferencing. Utilization of administrative assistance personnel.

4. Information/Data/User Interfaces (15%) Skill Level 2

Definitions: data processing systems, management information systems, decision support systems and management support systems. Work flow/information flow. User interfaces with support systems. Efficient versus effective information management.

5. Analysis and Design (20%) Skill Level 3

Productivity, organizational and political considerations and feasibility. Personnel and management support considerations. Productivity and cost-benefit analyses. User needs analysis. Training programs. Human and political factors. Organizational change and managerial effectiveness.

6. Future Office Systems (10%) Skill Level 2

Technological projections. Electronic and communications trends. Multi-functioning systems. Integrative support systems. Personnel development. Quality of work life issues. Limits and alternatives.

COURSE APPROACH

This course is designed to assist the computer information systems student in understanding business support systems--their design, organization and administration. Special attention is given to technological advances in data processing and office systems as well as to human and political considerations. While the course contains an overview of technologies associated with these systems, the thrust of the course is people and the impact of technology on management information and decision support systems.

The approach of this course should not be solely towards emphasizing those areas where data processing makes its contribution to the automated office, although this contribution ranges from simple, user-friendly menus to complex database information retrieval and analysis systems, as well as a wide range of office technologies which are computer driven. The information systems student should, at the same time, gain an understanding of the problems and concerns of those faced with daily management of the office, both in quantitative and qualitative aspects. The student should develop facility in systems analysis and design, conditioned by the user's needs and capabilities.

While there are general texts available for office systems, no text directly addresses the concerns presented above. It is suggested that instructors use journals, professional trade articles and commercial reference manuals for presentation of concepts and technologies. Both theoretical and actual case models should be presented and evaluated. Two or three case analyses may be assigned as out-of-class projects.

SELECTED REFERENCES

References will include major office and data processing journals, as well as trade magazines. Annotated references for selected texts will also be necessary. This aspect of the course will need constant attention, as the general area of automated and multi-functioning office systems present one of the fastest growing challenges to data processing. It is doubtful if the list of references will stay the same from one semester to the next.

Administrative Management

Data Management

Datamation

Information Systems News

Infosystems

Infoworld

Mini-Micro Systems

Modern Office Procedures

CIS-10 DECISION SUPPORT SYSTEMS

COURSE DESCRIPTION

An analysis of the highest level of information support systems which serve the manager user. This system provides quantitative-based information derived from one or more data bases within and/or external to an organization and used to aid managers in the decision-making process. Theoretical concepts will be applied to real-world applications with an analysis of examples from specific organizations. Prerequisite: CIS-1, Introduction to Computer-Based Systems.

STUDENT OUTCOMES

To develop an understanding of the area of decision support systems within organizations with reference to concepts and their application. To understand the information structure levels within organizations which lead to adoption of the decision support system concept. To gain an organization perspective of decision support and an approach for designing support systems. To be able to analyze alternate methods for designing, developing, and implementing support systems. To be able to analyze alternative technologies used for decision support systems.

COURSE CONTENT

1. Systems and Information Concepts (10%) — Skill Level 2

   General systems and information concepts and theory. Systems classifications: open, closed, deterministic and probabilistic. Data versus information. Value of information. Economics of information. Filtering and compression.

2. Structure of Systems in Respect to Organizations (10%) — Skill Level 2

   Humans in the systems process. Information and/or data as an informational resource in the organization. Systems plan and organizational strategy planning. Information and systems architecture for the organization.

3. Tools in Using Systems (15%) — Skill Level 2

   Models for systems representation. Model formulation theory. Graphical methods of analysis. Simulation concepts with distribution and sampling. Performance evaluation for systems refinement and optimization. Decision-making concepts: uncertainty, risk analysis and cost-benefit analysis.

4. The Information System for Management (15%) — Skill Level 2

   Taxonomy of operational information systems, data base management systems and management-related information systems to include definition, structure, need, identification, interface (man-machine), project management and development methodologies.

5. Decision Support Systems (25%) Skill Level 2

Taxonomy of decision support systems. Information definition. Requirements for decision making. External data base and decision support data base. Extraction and transformation methods. Defining decision models with examples. Developing the prototype DSS. Application using higher-level languages and packages as tools. Risk taking versus lost opportunities. Optimization of design through progressive cycles.

6. Application of Concepts with Case Studies (15%) Skill Level 3

Using examples from the literature and businesses, design and implement decision-making systems either with real-world or contrived case situations.

7. Communication Skills (10%) Skill Level 3

Presentation of research-related project in user and/or computer specialist mode. Write and/or access program using files in a decision mode. Calculation of cost/benefit concepts and analysis.

COURSE APPROACH

The course focuses on managers' decision support activities and the use of computer-based technology to support those activities in complex unstructured and semi-structured tasks. It covers potential benefits of computer support for decision making and stresses the decision maker's need for accurate data, external data and fast and timely perspectives of decision support in a dynamic environment. The organizational perspective of decision support is featured along with how the decision support system fits into the general MIS structure.

Emphasis throughout the course is to have the students exposed to the emerging area of decision support systems.

The larger framework of information systems will be covered from which both the MIS and DSS can be related and considered in both hypothetical and specific organizations.

Library and organizational research on the information systems structure will be completed by students.

SELECTED REFERENCES

Alter, Steven L. Decision Support Systems. Reading, Massachusetts: Addison-Wesley Publishing, 1980.

Keen, Peter F. and Scott-Morton, Michael S. Decision Support Systems. Reading, Massachusetts: Addison-Wesley Publishing, 1978.

Murdick, Robert G. MIS: Concepts and Design. Englewood Cliffs, New Jersey: Prentice-Hall, Inc., 1980.

Riley, M. J. Management Information Systems. 2nd edition. San Francisco, California: Holden-Day, Inc., 1981.

Schoderbek, Charles, Schoderbek, Peter, and Kefalas, Asterios. Management Systems: Conceptual Considerations. Dallas, Texas: Business Publications, Inc., 1980.

CIS 11 - ADVANCED DATABASE CONCEPTS

COURSE DESCRIPTION

Investigation and application of advanced database concepts including Database Administration, database technology and selection and acquisition of database management systems. In-depth practicum in data modeling and system development in a database environment. Overview of future trends in data management. Prerequisite: CIS-6, Database Program Development.

STUDENT OUTCOMES

To gain in-depth understanding of data administration and system development in a database environment. To develop understanding of the various roles within the data administration function of an organization. To gain a framework for identifying data administration issues and for resolving them. To understand and appreciate DBMS hardware and software requirements. To develop a framework for selecting and acquiring commercial DBMS's. To acquire a methodology for data modeling of the conceptual, storage and external schemas. To gain insight into the evolution and future trends in DBMS.

COURSE CONTENT

1. Database Administration (30%) — Skill Level 3

   Roles - Database Administrator, Database Technician, Application Programmer, Systems Analyst/Designer. Issues - security, privacy, integrity, concurrency control, backup, recovery, auditability and distributed processing.

2. Data Analysis, Design and Implementation Life-Cycle (30%) — Skill Level 3

   Requirements analysis - modeling the domain of information, data element analysis, user views and policy. Logical object analysis. Relational analysis. Aggregate design. Storage and access design. External schema documentation. Schemas - conceptual, logical, physical and external.

3. Database Technology (15%) — Skill Level 2

   Hardware - storage devices, back-end machines and database machines. Software - operating system interface, physical vs logical I/O and data communications. DBMS utilities. Performance monitoring.

4. Selection and Acquisition of a DBMS (15%) — Skill Level 2

5. Future Trends in DBMS (10%) — Skill Level 2

COURSE APPROACH

The approach to this course is built around a case study that provides an organizational setting as follows: The organization's data processing needs have grown to a point where management is considering implementation of a staff data administration function. An assignment for the student is to define the staffing requirements and responsibilities for a Database Administration department. Next an investigation of the issues surrounding a DBMS would be presented indicating the organizational impact of automated data management.

Playing the role of system developer, the student will investigate the case using a methodology directed at producing several models of the organization's data needs. These will include: a conceptual or logical model that represents a coherent, non-redundant, global view that supports all current user views and some specific sub-set of future user views; an internal or storage schema that provides for the physical mapping of data onto storage devices; and an external schema that presents each individual user's view.

The case should provide a realistic setting for some form of automated information processing including a hardware configuration. This would be analyzed in light of current storage technology. With the product of the data modeling exercise and an understanding of current technology, a hardware/software investigation should be mounted leading to a recommendation for a specific set of commercially available DBMS's suitable for acquisition.

The course is summarized with a brief history of data sharing and an overview of where things are going in data management.

SELECTED REFERENCES

Atre, Steve. Data Base: Structured Techniques for Design, Performance, and Management. New York: John Wiley and Sons, Inc., 1980.

Borkin, Sheldon A. Data Models: A Semantic Approach for Database Systems. Cambridge, Massachusetts: MIT Press, 1980.

Flavin, Michael. Fundamental Concepts of Information Modeling. New York: Yourdon Press, 1981.

Hutt, A. T. A Relational Database Management System. New York: John Wiley and Sons, Inc., 1979.

Johnson, Leroy F. and Cooper, Rodney H. File Techniques for Database Organization in COBOL. Englewood Cliffs, New Jersey: Prentice-Hall, Inc., 1981.

Katzan, Harry. Computer Data Management and Data Base Technology. New York: Van Nostrand Reinhold, 1975.

Kroenke, David. Database Processing: Fundamentals, Modeling, Applications. Palo Alto, California: Science Research Associates, Inc., 1977.

Lyon, John K. The Database Administrator. New York: John Wiley and Sons, Inc., 1976.

Martin, James. Computer Database Organization. 2nd edition. Englewood Cliffs, New Jersey: Prentice-Hall, Inc., 1977.

Meldman, Monte J., McLeod, Dennis J., Pellicore, Robert J., and Squire, Morris. RISS: A Relational Data Base Management System for Minicomputers. New York: Van Nostrand Reinhold, 1978.

Olle, T. William. The CODASYL Approach to Data Base Management. New York: John Wiley and Sons, Inc., 1978.

"Selection and Acquisition of Data Base Management Systems," a report of the CODASYL Systems Committee, March, 1976.

Tsichritzis, Dennis C. and Lochovsky, Frederick H. Data Base Management Systems. New York: Academic Press, 1977.

Ullman, Jeffrey D. Principles of Database Systems. Rockville, Maryland: Computer Science Press, 1980.

CIS-12 DISTRIBUTED DATA PROCESSING

COURSE DESCRIPTION

The features of centralized, decentralized and distributed systems will be examined. The impact of distributed systems on the business enterprise will be exposed via the medium of case studies. Technology implications of computer hardware, software and communications are discussed as they relate to the design, development and implementation of distributed data processing systems. Prerequisite: CIS-6, Database Program Development.

STUDENT OUTCOMES

To develop an understanding of the differences between centralized, decentralized and distributed data processing systems, their relationships with the business enterprise, data communications and the parameters affecting the implementation of the system. To gain the background for analysis, design, selection and evaluation of hardware, software and support required for a distributed data processing environment.

COURSE CONTENT

1. Fundamental Concepts of Distributed Processing (10%) — Skill Level 2
   Centralized, decentralized and distributed systems. Advantages and disadvantages of each configuration. System components.

2. Application Areas of DDP (5%) — Skill Level 3
   Design of computer systems related to the structure of the organization.

3. Hardware and Software (10%) — Skill Level 3
   Mini/micro computers. Terminals. Host operating systems. Access software. Word processing.

4. Data Communications Principles (10%) — Skill Level 2
   Basic concepts. Communications channels. Error control procedures.

5. Distributed Processing Networks (20%) — Skill Level 2
   Structures. Communication switching and routines. Network protocols. Communications channels. Computer networks. Tariffs and costs.

6. Distributed Database Structures (10%) — Skill Level 3
   Evolution of database systems. Database requirements. Database configurations.

7. Implementation and Management (15%) — Skill Level 3
   Project planning. Hardware evaluation. Software evaluation. Measurement and control. Security considerations (physical and system).

8. Case Studies (20%) — Skill Level 3
   Retail, manufacturing, etc. Systems modeling using simulation.

COURSE APPROACH

This course is primarily a discussion course in which the student gains an understanding of what DDP is and what its advantages and disadvantages are. Data communications concepts are discussed but not at an operational level of detail.

SELECTED REFERENCES

Booth, Grayce M. The Distributed System Environment. New York: McGraw-Hill, 1981.

Booth, Taylor L. Digital Networks and Computer Systems. New York: John Wiley and Sons, 1978.

Doll, Dixon R. Data Communications. New York: Wiley Interscience, 1978.

Fitzgerald, Jerry and Eason, Thomas. Fundamentals of Data Communications. New York: John Wiley and Sons, 1978.

Housley, Treror. Data Communications and Teleprocessing Systems. Englewood Cliffs, New Jersey: Prentice-Hall, Inc., 1979.

Martin, James. Communications Satellite Systems. Englewood Cliffs, New Jersey: Prentice-Hall, Inc., 1978.

Martin, James. Future Developments in Tele-Comm. Englewood Cliffs, New Jersey: Prentice-Hall, Inc., 1977.

Martin, James. Systems Analysis for Data Transmission. Englewood Cliffs, New Jersey: Prentice-Hall, Inc., 1972.

Martin, James. Telecommunications and the Computer. Englewood Cliffs, New Jersey: Prentice-Hall, Inc., 1976.

McGlynn, Daniel R. Distributed Processing and Data Communications. New York: John Wiley and Sons, 1978.

McNamara, John E. Technical Aspects of Data Communication. Bedford, Massachusetts: Digital Press, 1978.

Thierauf, Robert J. Distributed Processing Systems. Englewood Cliffs, New Jersey: Prentice-Hall, Inc., 1978.

CIS-13 EDP AUDIT AND CONTROLS

COURSE DESCRIPTION

An introduction to the fundamentals of EDP auditing. Emphasis on EDP controls, types of EDP audits, and concepts and techniques used in EDP audits. Exposure to risk assessment and professional standards in the field of EDP auditing. Prerequisite: CIS-6, Database Program Development.

STUDENT OUTCOMES

To develop an understanding of the EDP audit environment and processes. To realize the importance of EDP controls and the effect poor controls can have within a computer-based information system. To gain an understanding of the various kinds of audits performed of computer-based information systems and operations. To gain an appreciation of and motivation for proper data processing practices and management.

COURSE CONTENT

1. EDP Audit Environment and Computer Information Systems (10%) — Skill Level 3
   The EDP audit environment and its relationship with and effect on computer-based information systems. Relationships between the internal audit function, the external audit function, the public accounting function and the information systems function. EDP audit definitions. Discussion of major examples of computer abuse and their impacts on the business community.

2. Information Systems Controls (25%) — Skill Level 3
   Types of information systems controls: application controls, system development controls, information processing facility controls and horizontal control vs vertical controls. Preventive, detective and corrective controls. Controls and security.

3. Computer Audit Techniques (30%) — Skill Level 2
   Types of EDP audits: audits of applications, audits of systems development, audits of information processing facilities and SAS-3 reviews. Computer-assisted audit techniques such as test decking, integrated test facility, parallel simulation, system control audit review file, sample audit review file, snapshot, extended records, etc. Uses of audit software to verify results: confirmation, compariaon with file or physical and edit and reasonableness tests. Advantages and disadvantages of computer-assisted audit techniques.

4. Auditing Advanced Information Systems (20%) — Skill Level 2
   Techniques used to audit advanced systems which utilize a combination of any one of the following information processing techniques: on-line, real-time, teleprocessing, telecommunication, distributed processing, minicomputer, microcomputer, data bases, etc. Techniques used to audit data base systems. Cost of advanced controls. Audit technical expertise needed. Examination of minicomputer and microcomputer applications and environment.

5. Systems Approach to Auditing (15%) Skill Level 3
Concept and application of risk assessment. Concept and application of threat analysis. Concept and application of cost/benefit analysis in analyzing exposures and recommending controls.

COURSE APPROACH

This course provides fundamental knowledge of the EDP audit environment and processes. Although the course requires fundamental knowledge of accounting principles and processes and auditing concepts, these are used in the course to benefit and expand the knowledge of the information systems specialist.

Emphasis throughout the course should be on the importance of EDP controls, EDP audit reviews and their interaction with the business organization, especially the information systems department. The course will cover EDP control and computer audit techniques. The methodology used to introduce these topics is left to the discretion of the instructor; however, it should be representative of the current philosophy toward the examination of EDP controls and use of computer audit techniques.

There should be sufficient opportunity for limited research and moderate application of some of these concepts and practices through computer laboratory exercises, case studies and research papers. Included in the exercises should be requirements to design and develop an audit software program to verify results of a business application (e.g., general ledger system, payroll system, inventory control system, and the like).

There should be ample opportunity for students to accomplish two projects during the course. The first project should be a research paper on an EDP audit-related topic or a test data case could be used to provide hands-on experience. The second project should require students to apply the concepts and practices learned through a case study involving the use of an audit retrieval language, or, if one is not available, through use of a COBOL program written especially for that purpose.

SELECTED REFERENCES

Books:

Burch, John C., Jr. and Sandinas, Joseph L., Jr. Computer Control and Audit: A Total Systems Approach. New York: John Wiley and Sons, 1978.

Davis, Gordon B., Schaller, Carol A., and Adams, Donald L. Auditing & EDP. 2nd edition. New York: American Institute of Certified Public Accountants, 1981.

Mair, William C., Wood, Donald R., and Davis, Keagle W. Computer Control and Audit. Altamonte Springs, Florida: Institute of Internal Auditors, Inc., 1976.

Porter, W. Thomas and Perry, William E. Controls and Auditing. 3rd edition. Boston, Massachusetts: Kent Publishing Co., 1981.

Weber, Robert A. G EDP Auditing: Conceptual Foundations and Practice. New York: McGraw-Hill, 1981.

Case Studies:

"Case Studies in Computer Control and Auditing." Funded by The Touche Ross Foundation, June, 1978.

"Fedco Case Study". Frederick Gallegos, U.S. General Accounting Office, Los Angeles, California.

Publications:

"Audit and Evaluation of Computer Security," National Bureau of Standards Publication 500-19, 1977.

"Audit and Evaluation of Computer Security II: Systems Vulnerabilities and Controls," National Bureau of Standards Publication 500-57, 1980.

Computer Audit Guidelines. Canadian Institute of Chartered Accountants, 250 Bloor St. E., Toronto, Canada, 1975.

Computer Control Objectives-1980. EDP Auditor's Foundation, 373 Schemle Rd., Carol Stream, IL 60187.

EDP Auditor's Journal, EDP Auditor's Foundation, 373 Schemle Rd., Carol Stream, IL 60187.

EDP Auditor's Update. EDP Auditor's Foundation, 373 Schemle Rd., Carol Stream, IL 60187.

Management, Control, and Audit of Advanced EDP Systems. New York: American Institute of Certified Public Accountants, 1977.

Supplementary Audit Standards for Auditing Computer- ased Systems. U.S. General Accounting Office, 441 G. St., N.W., Washington, D.C. Attn: Special Publications.

Systems Auditability and Control -- Audit Practices. Altamonte Springs, Florida: Institute of Internal Auditors, 1977.

CIS-14 INFORMATION SYSTEMS PLANNING

COURSE DESCRIPTION

An introduction to the financial, technical and strategic information systems planning processes. Emphasis on the relationship of the information systems planning process to the overall business goals, policies, plans, management style and industry condition. Emphasis on the means of selecting large systems projects; assessing the installation's current state; determining processing, staffing, software, hardware and financing approaches. Review of hardware, software and services information sources. Prerequisite: CIS-7, Applied Software Development Project.

STUDENT OUTCOMES

To develop an understanding of the information systems planning process and how it relates to the overall business planning cycles. To provide an awareness of the decision making process in the selection, planning and approval of major computer projects and in determining processing philosophies. To develop an understanding of alternate methods of developing or acquiring hardware, software and human resources to meet company information needs. To acquire practice in researching and configuring a computer system and in presenting written and oral reports, justifications and recommendations.

COURSE CONTENT

1. Planning Cycle (15%) — Skill Level 2
   Business planning processes: operational, tactical and strategic. Impacts of rapid expansion versus conservative growth, centralized versus decentralized management, authoritarian versus democratic decision making and other business styles which impact the nature of the business and information systems plans. Developing the initial information systems plan and applications portfolio for a firm. Refining the plan to take advantage of changes in the business, advances in technology and the previous information systems successes or failures.

2. Selecting Systems Projects (10%) — Skill Level 3
   Methods of identifying and selecting major systems projects based on management style, technical feasibility and cost-benefit considerations. Identifying key decision makers and the organization's most functionally important or financially attractive projects. Formation of an executive team to obtain concurrence in larger organizations. Problems in establishing realistic priorities, budgets and target dates for small and large projects. Data base and data communications issues, and prerequisite applications for other systems, impacting project priorities.

3. Application Software Planning (15%) Skill Level 3

General tradeoffs for in-house developed applications versus packaged software. Importance of practical EDP standards. Identification of data base and data communications requirements, the complexity and rate of change of the business functions to be automated and the capabilities of the current information systems staff. Estimating the cost of developing software. Cost and difficulty of changing packaged software. Package documentation, on-going support and training and installation considerations. Hardware requirements of packages, and compatibility with data base/data communications software already installed. Locating proprietary software.

4. Staffing Planning (10%) Skill Level 3

Programmer, systems analyst and project manager productivity. Assessing the staffing requirements and training needs. Recruiting and retention considerations. Use of contract programming and consulting firms to supplement staff. Impact of software make or buy decisions on staff size. Application support and maintenance requirements. Datacom/data base and operating systems support requirements. Other staffing areas impacted by the systems plan. Work environment considerations. Development aids and management control requirements. Productivity improvement aids.

5. Hardware Planning (15%) Skill Level 3

Capacity sizing of key computer components: CPU, memory, network architecture, input/output devices, data entry devices and terminals. Data communications lines, protocols and diagnostic considerations. Other equipment requirements such as computer room expansion; fire suppression systems; physical security systems; electrical and air conditioning equipment; uninterruptible power supply systems; noise and safety considerations and work flow and staffing interaction considerations. Job accounting and performance monitoring functions. Impact of business growth on hardware plans. Hardware conversion and lead time issues. Vendor responsibilities and support. Supplies usage and control.

6. Acquisition Considerations (15%) Skill Level 3

Overview of hardware source alternatives (hardware vendors, third party leasing and sales, and company to company direct sales). Financing alternatives. Software rental, purchase and leasing alternatives. Software acquisition considerations, including compatibility with data base, data communications and hardware; maintainability of source code; conditions for acquiring future enhancements; documentation quality; training quality and quantity; competence of vendor personnel and their responsiveness to problems. Software and hardware price negotiation strategies. Contract negotiation processes. Preparing requests for procurement proposals. Maintaining control over consultants and hardware vendors.

7. The Decision Making Process (10%) Skill Level 3

The roles of the information systems manager, computer operations manager, software manager, project manager, application support manager, data communications manager and data base manager. Roles played by individual managers, project steering committees, maintenance steering committees and senior management advisory committees. Presentation strategies and methods. Determining decision criteria for tangible and intangible objectives. Selecting projects, defining project scope, identifying risks, qualifying estimating assumptions, identifying prerequisite projects, payback versus risk and small versus large projects. Other selection and prioritizing criteria; business needs, business changes, politics, technology changes, skill turnover and project and/or hardware/software disasters.

8. Communications Skills (10%) Skill Level 3

Review of information sources such as Datapro, Auerbach, ICP, vendor catalogs and brochures, consultants, industry associations, journals, trade magazines, trade shows, site visits and other information sources. Identifying the formal and informal policies and procedures relating to capital expenditures, approval processes, planning and budgeting procedures, auditing and follow-up practices and other issues which impact the method of presentation and reporting progress. Written and oral progress reports. Conducting meetings. Matching reporting styles to target audience and subject matter.

COURSE APPROACH

This course is designed to give the student a basic understanding of the many factors that come into play in systems planning and to provide an appreciation of the multiple roles to be played as the student's career progresses through various information systems job functions.

At the outset of the course it should be emphasized that the type of planning and amount of effort spent varies greatly by the organization size and by the level for which the planning is being done. The course should demonstrate that the business style, plans and goals drive the information systems planning process, and organizational decisions impact the scope of the planning function considerably. It should be emphasized that development of information systems plans is not a one time activity, but rather a cyclical process with varying time horizons.

Some time should be spent in making the student aware of the various tools, techniques and reference sources available in the areas of hardware selection, financing, software make or buy decisions and, by use of case histories, students should be able to practice selecting and presenting solutions using the reference materials. There should be ample opportunity for the student to practice written and oral communication skills, with emphasis particularly on ease of understanding, crispness, clarity, specificity and use of traditional business terminology so that users and managers alike can understand it.

Each student should complete at least the following projects: A case study on the selection of hardware; a case study on the selection of a software application package; a critique of a faulty contract for software and hardware acquisition; a number of relatively simple financing alternative problems; a written three to five year plan based on a case history, and a presentation of the plan to the class. This latter project may be done in a team environment rather than on an individual basis, but it should tie together all of the course material.

SELECTED REFERENCES

Auerbach DATAWORLD (4 Vols.) for equipment, applications software, data communications and mini-computers. (Auerbach Publishers, Inc., 6560 North Park Drive, Pennsauken, NJ 08109.)

DATAPRO/70 (3 Vols.), for evaluation, selection and pricing of equipment, systems software, mini-computers, data base/data communications alternatives, software, productivity aids and supplies. (DATAPRO Research Corp., 1805 Underwood Blvd, Delran, N.J. 08075.)

ICP Software Directory (6 Vols.) and ICP Interface, published quarterly for packaged applications and systems software pricing and features. (International Computer Programs, Inc., 9000 Keystone Crossing, Indianapolis, IN 46240.)

IBM Configuration Summaries, to be used for case studies.

Various vendor supplied catalogs and brochures of software and hardware selection and pricing.

CIS-15 INFORMATION RESOURCE MANAGEMENT

COURSE DESCRIPTION

A seminar course providing a broad overview of the information systems management function. The course emphasizes information systems management, with particular attention on planning, organizing and controlling user services and managing the computer information systems development process. Coverage of subject matter through lectures, readings, discussions and case study analysis. Prerequisite: CIS-7, Applied Software Development Project.

STUDENT OUTCOMES

To gain useful insights for relating the activities of programming and systems analysis and design to a set of management requirements. To gain awareness of the basic literature in the subject area of information systems. To gain awareness of key elements in managing the information systems function. To build a framework for selecting and pursuing a career in information systems through awareness of the underlying management process and of future developments in the information field.

COURSE CONTENT

1. Introduction and Review (5%) Skill Level 3
   Review of the management process. How the organization functions: role of strategic, functional and line organizations. Role of staff functions. Central management tasks: planning, organizing, resource allocation and control. Role of Information Resource Manager. Organizational hierarchy for the information systems function. Management reporting: role of information in successful management, need for measurements, need for decision support information, need for information to satisfy legal criteria and government requirements.

2. Stages of Computer Information Systems Growth (10%) Skill Level 3
   Overview of the evolution of a business organization. Organizational growth and the need for and characteristics of information. Evolution of computer information systems: stages of EDP growth from cost-reduction activities to complex database applications, decision support systems and communications networks. Aspects of organization, planning and control throughout information systems evolution. Evolution of the applications portfolio. Evolution in the role of information systems relative to management of the enterprise. The organizational learning curve: evolution in user sophistication and awareness.

3. Information Systems Management (20%) Skill Level 3

Sociology of the information systems organization: (DP vendors, CIS personnel, mission-oriented personnel, resource allocators) their natural self-interests, alliances and differing views of information systems. Business planning and control: processes for managing computer information systems over a strategic time horizon. Resource planning and control: process of tactical resource allocation and administration. Service level definition and control. Change control. Production and distribution of information. Administration: budgeting, accounting, chargeout, education and training and management reporting. User needs. Measures of overall success of information systems.

4. Organization and Control (10%) Skill Level 2

Organizational structures: division of labor, mechanisms for coordination of activities and shapes of organizations. Organizational and technological aspects of centralization, decentralization and distribution. Views of distributed processing. Motivation for distribution. Distribution of processes, data and control. Internals and externals of information systems organization: structure and reporting arrangements.

5. Information Systems Development (20%) Skill Level 2

Framework for systems development: key success factors, risk reduction via formal planning tools, formal control tools, internal integration and external integration. The descriptive systems life cycle, identifying life-cycle stages, activities and interactions that occur during systems development. Project management concepts and the use of project teams: staffing, planning, organizing, and controlling project teams. Methodologies of systems development.

6. Legal Considerations in Computer Information Systems (5%) Skill Level 2

Contracts, patents, copyrights, trade secrets and unfair competition. Compatibility of equipment and software. Liability theories, warranties and contract remedies. Delivery. The nature of contracts. Hardware products and services: trade-ins, allowances, concessions and comparative contract clauses. Vendor-supplied software: comparative contract clauses. Customized software and hardware packages. Time-sharing and service bureau products and services. Legal aspects of hardware and software maintenance. Privacy legislation. Data security and integrity: elements of computer fraud.

7. Communications Skills (10%) Skill Level 3

Elements of communications skills. Ability to influence others. The systems analyst as a change agent. The nature of professionalism: acquisition and maintenance of skills. The nature of success, failure, frustration, and pressure in information systems fields. Professional societies. Methods for using and sharing technical information. Presentation of technical research: written reports, oral presentations and defense of research and interpretations. Assessing technical presentations.

8. Personnel Issues (10%) Skill Level 2
Career paths in computer information systems. Sources of personnel. Skill requirements. Training sources. Employee motivation. People management: setting objectives, performing appraisals, counseling and people measurements. Aspects of reward and punishment systems. Computer information systems career requirements: academic preparation, continuing education, on-the-job learning and performance.

9. Future Directions of Computer Information Systems (10%) Skill Level 2
Directions in hardware technology, software development techniques, and user demand for applications in inquiry and decision support systems, coupled with growing user sophistication. Office automation and its relationship to computer information systems. Societal issues of automation: potential impact on clerical, operational and management roles. Role of the computer information systems practitioner in future environments.

COURSE APPROACH

This seminar course is to be used to give students a broad overview of emerging issues in computer information systems, particularly as viewed from the vantage point of the information systems manager. Students should be exposed to concepts, terminology and sources of information related to managing the information systems function of a business enterprise. The course should feature individual projects wherein the student can research one of the topical areas and write a formal report, giving an oral briefing to the class.

The course should help the student develop answers to such fundamental questions as What is CIS management? What does a CIS manager do? What are my career choices in CIS? What is success and failure in CIS management? What will be my job responsibilities? What pressures will I face?

The course focus should be on managing the system's resources: people, hardware, budgets, applications portfolios, software development, and the like, with emphasis on information systems management within the total organizational context and how elements of the computer information systems function fit together as an integrated system.

There should be two or three case studies assigned. They should be assigned near the end of the class, after a reasonably complete level of understanding has been gained of the information systems function.

SELECTED REFERENCES

Books:

An Architecture for Managing the Information Systems Business, Volume 1: Management Overview. GE20-0662. White Plains, New York: IBM Corporation, 1980.

Bernacci, Richard L. and Larsen, Gerald H. Data Processing Contracts and the Law. Boston, Massachusetts: Little, Brown and Company, 1974.

Bigelow, Robert and Nycum, Susan B. Your Computer and the Law. Englewood Cliffs, New Jersey: Prentice-Hall, Inc., 1975.

Brandon, Richard. Data Processing Organization and Manpower Planning. New York: Petrocelli Books, 1974.

Freed, Louis. Practical Data Processing Management. Reston, Virginia: Reston Publishing Co., 1979.

Kast, Fremont E. and Rosenzweig, James E. (Editors) Contingency Views of Organization and Management. Palo Alto, California: Science Research Associates, Inc., 1975.

Liebowitz, Burt H. and Carson, John J. Distributed Processing. New York: Institute of Electronic and Electrical Engineers Computer Society, 1978.

McFarlan, F. Warren and Nolan, Richard L. (Editors) The Information Systems Handbook. Homewood, Illinois: Dow Jones-Irwin, 1975.

McLean, Ephraim R. and Soden, John V. Strategic Planning for MIS. New York: John Wiley and Sons, Inc., 1977.

Miller, Arthur R. "The National Data Center and Personal Privacy," Hoffman Lance J. (Editor) Security and Privacy in Computer Systems. Los Angeles, California: Melville Publishing Co., 1973.

Mintzberg, Henry. The Structuring of Organizations. Englewood Cliffs, New Jersey: Prentice-Hall, Inc., 1979.

Ronstadt, Robert. The Art of Case Analysis. Dover, Massachusetts: Lord Publishing, 1980.

Sanders Donald H. and Birkin, Stanley J. (Editors) Computers and Management in a Changing Society. New York: McGraw-Hill, 1970.

Schaeffer, Howard. Data Center Operations. Englewood Cliffs, New Jersey: Prentice-Hall, Inc., 1981.

Case Studies:

"Decatur Industries, Inc." 9-180-010. Intercollegiate Case Clearing House. Harvard Business School.

"First American Bank (B)," 9-176-001. Intercollegiate Case Clearing House. Harvard Business School.

"Quality Life of New York," 9-179-200. Intercollegiate Case Clearing House. Harvard Business School.

Journals:

Ocassional articles, CISE (MIT Sloan School).

Ocassional articles, Harvard Business Review.